Entire Screen Builder

	Version 5.2.1	Installation and Configuration

software AG

This document applies to Entire Screen Builder Version 5.2.1 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

© Copyright Software AG 1999-2003
All rights reserved.

The name Software AG and/or all Software AG product names are either trademarks or registered trademarks of Software AG. Other company and product names mentioned herein may be trademarks of their respective owners.

Table of Contents

Installation and Configuration 1
Installation and Configuration 1
Supported Communication Methods 3
Supported Communication Methods 3
Telnet 3270(E) 3
Telnet VT 3
BS2000 TCP/IP 3
Natural UNIX 4
Possible Setup Scenarios 5
Possible Setup Scenarios 5
Development Environment 5
Production Environment 6
Installing the Entire Screen Builder Server and the SDK on a PC 7
Installing the Entire Screen Builder Server and the SDK on a PC 7
Prerequisites for Windows 8
Prerequisites for Reading the Online Documentation 9
License Files 10
Installation Dependencies 11
Installing/Updating Entire Screen Builder 11
Shortcuts in the Start Menu 15
Program Folders 16
Environment Variables 18
Registry 18
Using the Demo Application 18
Exporting Objects from Entire Connection to Entire Screen Builder 21
Installing the Entire Screen Builder Server in the UNIX Environment 22
Installing the Entire Screen Builder Server in the UNIX Environment 22
Differences between the Windows and UNIX Versions 23
Prerequisites 23
License Files 24
Setting Up the Entire Screen Builder Server 25
Step 1: Stop the Entire Screen Builder Server 25
Step 2: Establish the Environment and Install the Product 25
Starting and Stopping the Entire Screen Builder Daemon 27
Activating Entire Screen Builder at System Startup 28
Preparing the System V Style Startup Procedures for Entire Screen Builder 29
Changing the System Management Hub Startup Procedure for Entire Screen Builder 32
Entire Screen Builder Server Directories 34
Working with the Entire Screen Builder Server in the UNIX Environment 35
Deinstalling the Entire Screen Builder Server 35
Updating Manually Deployed Files on the HTTP Server 36
Updating Manually Deployed Files on the HTTP Server 36
Installing Natural for Entire Screen Builder on UNIX Hosts 37
Installing Natural for Entire Screen Builder on UNIX Hosts 37
Prerequisites 37
Setting Up the Entire Screen Builder Components 38
Step 1: Stop the Entire Screen Builder Daemons 38
Step 2: Establish the Environment 38

Step 3: Install Natural and Entire Screen Builder with the Demo Application 38
Step 4: Check the Environment Variables for Entire Screen Builder 39
Step 5: Read the READ_NSW Files 39
Directories 40
Configuration Files 40
Working with the Entire Screen Builder UNIX Components 41
Starting a New Natural Application 41
Starting and Stopping the Entire Screen Builder Daemon 43
Installing Natural for Entire Screen Builder on OpenVMS Hosts 44
Installing Natural for Entire Screen Builder on OpenVMS Hosts 44
Prerequisites 44
Setting Up the Entire Screen Builder Components 45
Step 1: Stop the Entire Screen Builder Daemons 45
Step 2: Establish the Environment 45
Step 3: Install Natural and Entire Screen Builder with the Demo Application 45
Step 4: Check the Environment for Entire Screen Builder 46
Step 5: Define the TCP Port Number 46
Directories 47
Configuration File 48
Setting Up and Activating the NSWSRVD Daemon 49
Customizing the Web Viewer 50
Customizing the Web Viewer 50
General Information 50
HTML Page 51
Id Attribute 51
Classid Attribute 51
Codebase Attribute 51
Param Tag 52
Other Attributes 52
Customizing the Terminal Viewer 53
Customizing the Terminal Viewer 53
General Information 53
HTML Page 54
Id Attribute 54
Classid Attribute 54
Codebase Attribute 54
Param Tag 55
Other Attributes 55
Making the Online Documentation Available to All Users 56
Using the Viewers with Natural on UNIX and OpenVMS Hosts 57
Using the Viewers with Natural on UNIX and OpenVMS Hosts 57
Logon Credentials 57
Closing the Natural Application and Natural in Error Situations 60
Restrictions 62
HTTP Tunneling 64
HTTP Tunneling 64
Setting Up Tunneling 64
Microsoft Internet Information Server under Windows 65
Apache Web Server 2.0 under Windows 68
Apache Web Server 1.3 and 2.0 on Solaris 70
Customizing the Viewers 72

Windows Viewer 73
Web Viewer 73
Terminal Viewer (Browser Version) 73
Configuring the XML Version 74
Configuring the XML Version 74
About the XML Version 74
Browser Restrictions 76
Netscape/Mozilla 76
Others 76
Basic Requirements for the XML Version 77
Required Folders on the Internet Information Server 78
Configuring the Internet Information Server 79
Configuring the ISAPI DLL 80
Overview of Options 81
Defined Websites 84
Browser Configuration 85
Customizing the XSL Files 87

Installation and Configuration

This documentation explains how to install and configure Entire Screen Builder.

You can install Entire Screen Builder either on a Windows or UNIX system. Please note that the full functionality is only available on Windows systems while there are several restrictions (platform availability, functionality) for UNIX systems.

The installation program installs and preconfigures the Entire Screen Builder Server on a Windows or UNIX machine. It also copies the client software for the different viewers to subdirectories on the server machine.

If the host system on which you run your Natural applications is a UNIX or OpenVMS system, additional software for Entire Screen Builder has to be installed on the host (i.e. a daemon which starts a Natural program that is linked with an Entire Screen Builder library).

If the host system on which you run your applications is a mainframe, you only have to install Entire Screen Builder. No additional software has to be installed on the host system.

The following topics provide all information required for installing and configuring Entire Screen Builder:

Supported Communication Methods	Information on the communication methods supported by Entire Screen Builder.
Possible Setup Scenarios	Diagrams which show the components and architecture for a development environment and a production environment.
Installing the Entire Screen Builder Server and the SDK on a PC	Prerequisites, installation instructions and information on license files, shortcuts in the Start menu, program folders, environment variables, and keys in the registry. How to use the demo application, and information on the export feature of Entire Connection.
Installing the Entire Screen Builder Server in the UNIX Environment	Differences between the Windows and UNIX versions, prerequisites, installation instructions, and directories. How to work with the Entire Screen Builder Server in the UNIX environment, and how to deinstall the Entire Screen Builder Server.
Updating Manually Deployed Files on the HTTP Server	Overview of files and modules that have to be modified on the HTTP server after an update installation.
Installing Natural for Entire Screen Builder on UNIX Hosts	Prerequisites, directories, installation instructions, configuration files, and how to work with the Entire Screen Builder UNIX components.
Installing Natural for Entire Screen Builder on OpenVMS Hosts	Prerequisites, directories, installation instructions, configuration file, and how to set up and activate the NSWSRVD daemon.
Customizing the Web Viewer	How to customize the example HTML pages for the Web Viewer and how make them available to all users in your environment.
Customizing the Terminal Viewer	How to customize the example HTML page for the browser version of the Terminal Viewer and how make it available to all users in your environment (including documentation).
Using the Viewers with Natural on UNIX and OpenVMS Hosts	How the logon credentials are picked up when logging on to UNIX and OpenVMS hosts. How to close the Natural application and Natural in error situations. Restrictions when using the viewers with Natural applications on UNIX and OpenVMS hosts.
HTTP Tunneling	How to set up tunneling for the supported HTTP servers, and how to enable tunneling for the different viewers.
Configuring the XML Version	Prerequisites, how to configure the XML Version and how to connect to the host.

Supported Communication Methods

Entire Screen Builder supports the following communication methods:

- Telnet 3270(E)
- Telnet VT
- BS2000 TCP/IP
- Natural UNIX

Telnet 3270(E)

Entire Screen Builder supports TCP/IP TN3270 and TN3270E communication.

You can use any network adapter that is supported by any TCP/IP stack software which provides the WinSock 2 interface.

The TCP/IP stack software must be installed and active in order to activate terminal emulation.

This mode supports extended attribute bytes (EABs).

See also: *Communication Properties for Telnet TN3270* in Entire Screen Builder's *System Management Hub* documentation.

Telnet VT

Entire Screen Builder supports VT100, VT220 and VT320 communication with any network adapter that is supported by any TCP/IP stack software which provides the WinSock 2 interface.

The TCP/IP stack software must be installed and active in order to activate terminal emulation.

The communication for the data transfer with Natural UNIX and Natural OpenVMS is done using an additional port. During installation the number of this port is entered in the **UNIX data transfer port number** property in the System Management Hub. See *Server Settings* in Entire Screen Builder's *System Management Hub* documentation.

See also: *Communication Properties for Telnet VT* in Entire Screen Builder's *System Management Hub* documentation.

BS2000 TCP/IP

This communication method emulates the standard 9750 terminal which is a 24 by 80 characters display without colors. Local printing is not supported.

In addition to the standard 9750 terminal features, the following features of the 975x family are supported:

- 80 FTZ per line
- 20 P-keys
- 24 F-keys
- reverse video
- full 9756-type memory support for P-Registers

In Natural environments, the color terminal type 9763 (7 bit) is also supported. As a prerequisite, Natural Version 3 or above must be installed. By default, Natural uses the terminal type 9750 (monochrome). To activate the terminal type 9763, use the following Natural terminal command (either in a screen or in a program):

```
%T=9763
```

When activating the terminal type 9763, it is recommended that you also load the Siemens function keys F1 through F20 using the following Natural terminal command:

```
%KN
```

Entire Screen Builder supports TCP/IP communication with BS2000 hosts with any network adapter that is supported by any TCP/IP stack software which provides the WinSock 2 interface.

The prerequisite on the host side is the communication subsystem BCAM version V.11, which establishes the connection with the host (available within the Siemens product DCAM).

No third-party software is needed for Entire Screen Builder to activate terminal emulation.

To make the terminal emulation key settings similar to those on a BS2000 keyboard, use the predefined key scheme BS2KEYS1. See *Key Schemes* in Entire Screen Builder's *System Management Hub* documentation.

See also: *Communication Properties for BS2000* in Entire Screen Builder's *System Management Hub* documentation.

Natural UNIX

This is a proprietary protocol for communicating with Natural applications on UNIX and OpenVMS hosts.

You can use any network adapter that is supported by any TCP/IP stack software which provides the WinSock 2 interface.

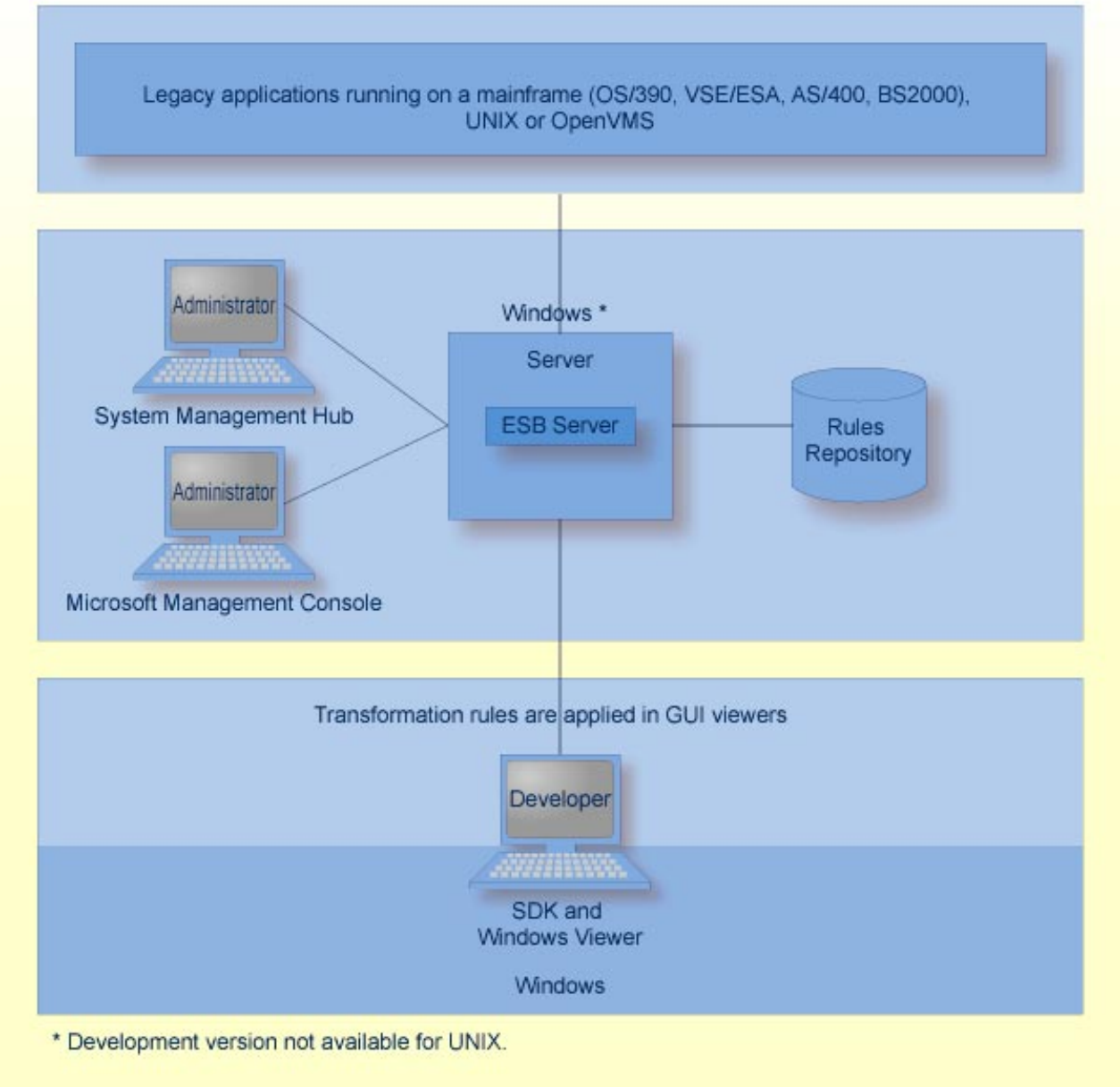
See also: *Communication Properties for Natural UNIX* in Entire Screen Builder's *System Management Hub* documentation.

Possible Setup Scenarios

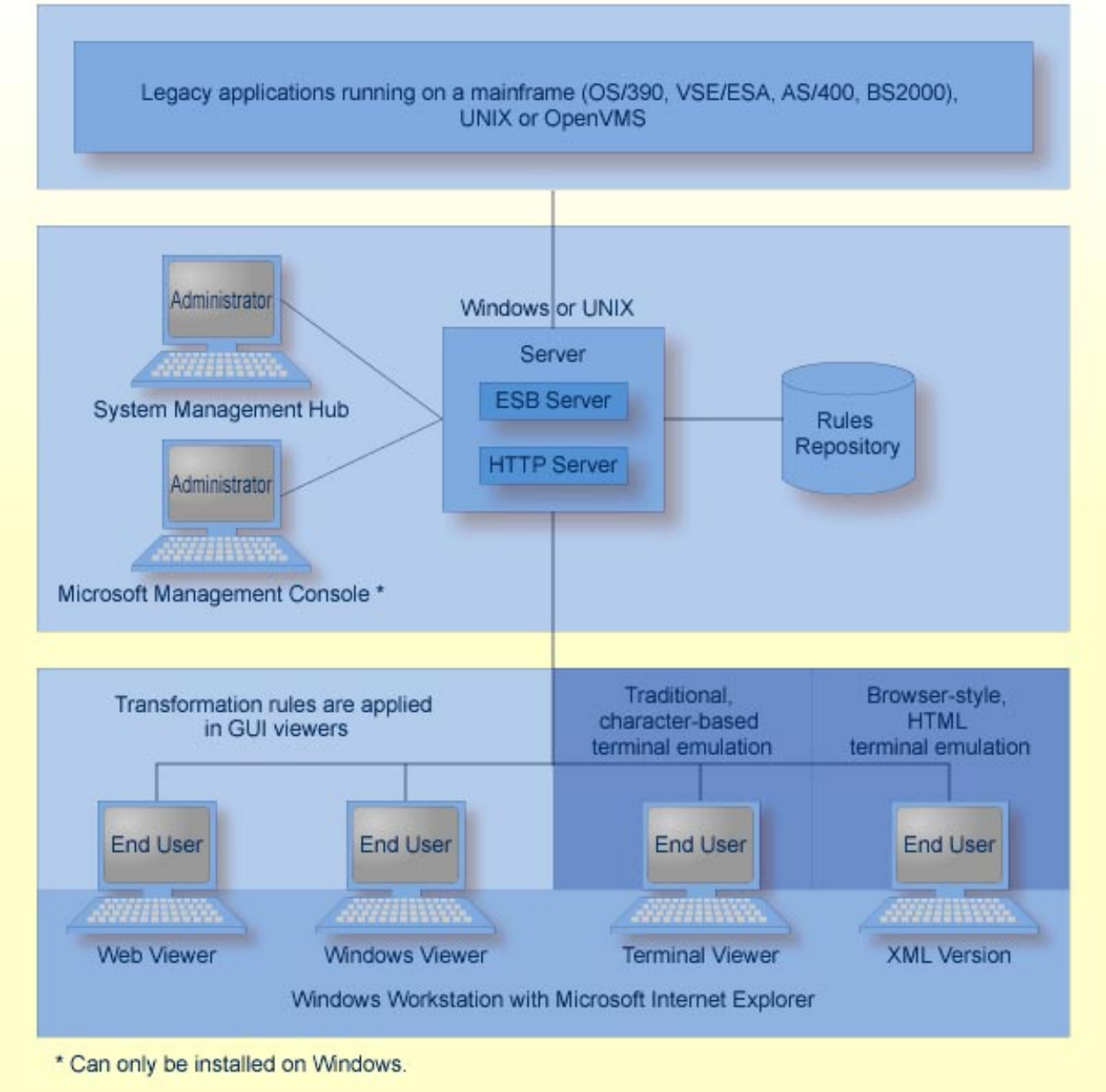
Entire Screen Builder can be installed in different configurations. The diagrams in this section show the components and architecture for the following scenarios:

- Development Environment
- Production Environment

Development Environment



Production Environment



Installing the Entire Screen Builder Server and the SDK on a PC

This chapter covers the following topics:

- Prerequisites for Windows
- Prerequisites for Reading the Online Documentation
- License Files
- Installation Dependencies
- Installing/Updating Entire Screen Builder
- Shortcuts in the Start Menu
- Program Folders
- Environment Variables
- Registry
- Using the Demo Application
- Exporting Objects from Entire Connection to Entire Screen Builder

---

Prerequisites for Windows

The following is required in order to install and run Entire Screen Builder on a PC:

Entire Screen Builder Server and SDK	The Entire Screen Builder Server and the SDK can be installed on the following operating systems: • Microsoft Windows 2000 Professional, Server or Advanced Server (Windows Clustering and Windows 2000 Datacenter are not supported), or • Microsoft Windows XP Professional (fast user switching is not supported), or • Microsoft Windows Server 2003 Standard Edition or Enterprise Edition. See also: *Installing the Entire Screen Builder Server in the UNIX Environment*.
System Management Hub	Software AG's System Management Hub Version 3.3.1.2 or above. The System Mangement Hub is automatically installed when you install Entire Screen Builder on a PC. For supported browsers, see the System Management Hub documentation.
Server for the XML Version	• Microsoft Internet Information Server 5, and • MSXML3 which is part of Microsoft Internet Explorer 6.0. See *Configuring the XML Version* for further information.
Presentation Clients	**Web Viewer and Terminal Viewer:** • A Windows PC with an operating system supported by Microsoft, and with Microsoft Internet Explorer 5 or 6. • Any HTTP server. An HTTP server is not required if you open the HTML page for a viewer from your local disk or LAN. **Windows Viewer:** • A browser is not required. • An HTTP server is not required. **XML Version:** • Microsoft Internet Explorer 5 or 6. Other browsers with restrictions; for detailed information, see *Browser Restrictions* in the section *Configuring the XML Version*.

Tunneling	For tunneling, you can use one of the following HTTP servers:
	• Microsoft Internet Information Server:
	  ○ Version 5.0 for Windows 2000
	  ○ Version 5.1 for Windows XP Professional
	• Apache Web Server:
	  ○ Version 2.0 for Windows.
	See *HTTP Tunneling* for further information.
	If tunneling is not used, you can use any HTTP server.
	Note:
	Tunneling prerequisites for Solaris are provided in the section *Installing the Entire Screen Builder Server in the UNIX Environment*

Prerequisites for Reading the Online Documentation

HTML Browser	For viewing the online documentation, a browser capable of supporting Java, JavaScript, and cascading style sheets is recommended. The documentation has been successfully tested with: • Microsoft Internet Explorer version 5.x or 6.x and the Microsoft Java Virtual Machine 5.0.0.3810. If you are using an older version of the Microsoft JVM, we strongly recommend you to upgrade it in order to avoid problems (such as the full-text search hanging up). By upgrading you also benefit from Microsoft's latest security enhancements. If you want to upgrade your Microsoft Virtual Machine, please visit the Microsoft web site (http://www.microsoft.com/). The browser is freely available for download.
PDF Reader	For viewing and printing the PDF documentation, Adobe Reader is recommended.

License Files

During installation, you have to specify the location of the file containing your Entire Screen Builder license key. You should have a license file that is provided by Software AG either on a storage medium such as a diskette in the Entire Screen Builder box or by E-mail. If you have received your license file by E-mail, save it to your local hard disk before starting the installation.

Note:
If you are installing Entire Screen Builder on a laptop and you have received your license file on a diskette, please note that some laptop configurations do not allow you access to the CD drive and the diskette drive simultaneously. In such cases, you must copy the license file to a location that is accessible while the CD drive is in use, such as your laptop's hard disk, before you start the installation.

Ensure that the file containing the license key is in a location that will be accessible during installation, such as on the file system or in a diskette drive. In general, Software AG recommends you to place the license file on the file system before starting the installation.

Depending on the license file you specify during installation, one of the following is installed (*nnn* in the license file name stands for the current version number):

License File	Description	Type of Environment
NSW*nnn*.xml	Entire Screen Builder SDK. Only available for Windows.	Development environment.
EWV*nnn*.xml	Entire Screen Builder GUI Version.	Production environment.
ETV*nnn*.xml	Entire Screen Builder Terminal Version.	Production environment.
EMV*nnn*.xml	Entire Screen Builder XML Version.	Production environment.

Note:
License files are not required for the standalone versions of the Terminal Viewer and Windows Viewer.

If you want to install all production environment versions, you have to install one version first and then load the license files for the subsequent versions using the System Management Hub. See *License Files* in Entire Screen Builder's *System Management Hub* documentation.

During runtime, the number of opened connections is compared with the number of users in the license file. The number of users must be equal to or greater than the number of opened connections on the target machine.

Installation Dependencies

A production environment version cannot coexist with a development environment version on the same PC (and vice versa). For example, if you want to install a production environment version on a PC on which a development environment version has already been installed, you have to uninstall the development environment version first.

The standalone version of the Windows Viewer can only be installed when an Entire Screen Builder Server has not been installed on the same machine. (Note that the Entire Screen Builder Server is always installed with a development environment version.)

When the standalone version of the Windows Viewer has already been installed, it is not possible to install a development environment version.

When the standalone version of the Windows Viewer has already been installed, it is not possible to install a production environment version including an Entire Screen Builder Server. A dialog box will appear during installation asking whether you want to continue. After that, the server installation will be disabled.

Installing/Updating Entire Screen Builder

In order to install Entire Screen Builder successfully, you must have administrator rights for your PC as the installation process requires access rights to system resources.

Important:
It is not necessary to install Entire Screen Builder as described below if you only want to install a production environment on a UNIX system.

The setup program checks for existing Entire Screen Builder Version 4.1.1, 4.2.1, 4.3.1 and 5.1.1 installations. If such an installation is detected, it is updated to the current version. In this case, the Entire Screen Builder Server is automatically stopped by the setup program. It is automatically restarted after a successful update. Before starting an update installation, it is therefore recommended that you inform all connected users (for example, with the Send Message function in the System Management Hub) and close all Entire Screen Builder viewers.

To install/update Entire Screen Builder

1. Close any active Windows applications.
2. Insert the Entire Screen Builder CD into your CD drive.

   The setup program is automatically started and guides you through the installation.

   If the automatic startup option is disabled on your system, execute *Setup.exe* which can be found in the root directory of the CD.
3. In the first installation dialog, choose **Install Products**.

A dialog appears in which you can choose one of the following options:

Option	Description
Production Version	Installs the production environment version which contains all modules for a multi-client production installation.
Development Version	Installs the development environment version which contains all modules for developing and testing the transformation rules for host screen GUIfication.
Terminal Viewer Standalone	Installs the standalone version of the Terminal Viewer.
Windows Viewer Standalone	Installs the standalone version of the Windows Viewer.

4. Choose the required option and follow the instructions in the resulting dialog box.

   Notes:

   1. With a custom installation, which is available for installing a production or development environment, all features can be installed separately.
   2. In a development environment, the SDK and the Entire Screen Builder Server are always installed. These features cannot be deselected in the dialog which appears for a custom installation.
   3. The feature "Entire Screen Builder Server" installs Software AG's System Management Hub which is used to administrate the Entire Screen Builder Server. See Entire Screen Builder's *System Management Hub* documentation for further information.
   4. The features "Extensions for the Internet Information Server for the XML Version" and "Extensions for the Internet Information Server for Tunneling" are only available when the Microsoft Internet Information Server (IIS) is installed. They are only available with a custom installation.
   5. When you install the extensions for the XML Version, a dialog appears asking whether you want to configure the ISAPI DLL. See *Configuring the ISAPI DLL* for further information.
   6. When you install the extensions for tunneling, a dialog box appears in which you have to enter parameters. This feature is only available for production environment versions. See *Microsoft Internet Information Server under Windows* in the section *HTTP Tunneling* for further information on this dialog box.
   7. When you install the standalone version of the Terminal Viewer, a dialog appears in which you have to specify the connection parameters for the machine on which the Entire Screen Builder Server is running. See *Defining the Entire Screen Builder Server* in the *Terminal Viewer* documentation.

5. After installation, you may have to reboot your computer. When this is required, the corresponding dialog box will appear asking whether you want to reboot now or later.

Notes for IIS 6.0 (Windows Server 2003):

The security settings for IIS 6.0 have been extended to allow the enabling or disabling of individual ISAPI extension DLLs. If you do not have the security options of IIS set to allow all ISAPI extensions, you must enable the Entire Screen Builder ISAPI DLLs that you need.

- **Tunneling**
  Tunneling is provided by *EsbIsapi.dll* contained in the virtual folder *esbscripts*.

- **XML Version**
  The XML Version requires the ISAPI extension DLL *nswxmlisapi.dll* contained in the virtual folder *esbscripts*.

Our installation process does not alter these settings to allow you to choose the method of security setting best suited to your individual requirements.

Production Environment on UNIX

If you want to install/update an Entire Screen Builder production environment on a UNIX system, proceed as described in the following section:

- *Installing the Entire Screen Builder Server in the UNIX Environment*

Natural UNIX and Natural OpenVMS

If your legacy applications run with Natural UNIX or Natural OpenVMS, proceed as described in the following sections:

- *Installing Natural for Entire Screen Builder on UNIX Hosts*
- *Installing Natural for Entire Screen Builder on OpenVMS Hosts*
- *Using the Viewers with Natural on UNIX and OpenVMS Hosts*

First-time Installation

After a first-time installation, it is recommended that the developer proceeds as follows:

1. Use the System Management Hub to define all host sessions. See *Host Sessions* in Entire Screen Builder's *System Management Hub* documentation.
2. Use the SDK to define all transformation rules. See the documentation *Defining the Rules Using the SDK*.
3. Invoke the Windows Viewer from the SDK to check whether the rules are applied as desired. See *Starting a Host Session* in the documentation *Defining the Rules Using the SDK*.

Make Viewers Available - Browser Versions

How to make the browser versions of the viewers available to all users in your environment is explained in the following sections:

- *Customizing the Web Viewer*
- *Customizing the Terminal Viewer*

Make Viewers Available - Standalone Versions

The standalone versions of the Windows Viewer and Terminal Viewer run as regular Windows applications on a client workstation. Therefore, they do not require a browser on the client workstation. To make these viewers available to all users in your environment install them using the setup program.

Update Installation

In the case of an update installation, proceed as described in the following section:

- *Updating Manually Deployed Files on the HTTP Server*.

Tunneling

How to use tunneling with the viewers, is explained in the following section:

- *HTTP Tunneling*

Make XML Version Available

How to make the XML Version available to all users in your environment is explained in the following section:

- *Configuring the XML Version*

Events

Entire Screen Builder events can be checked using the Windows Event Viewer.

Shortcuts in the Start Menu

The information in this section applies when you install a development environment version or a production environment version. It does not apply to the standalone versions of the Terminal Viewer and Windows Viewer. Shortcuts for the standalone versions are always available from the root of the Programs folder of the Windows Start menu

If you do not specify otherwise during installation, "Software AG Entire Screen Builder *n.n.n*" (where *n.n.n* is the version number) appears in the Programs folder of the Windows Start menu. This folder contains all shortcuts for invoking the Entire Screen Builder components.

The following table indicates the shortcuts that can be installed with a specific license:

Shortcut	Development Environment	Production Environment
Online Documentation	X	X
Readme	X	X
SDK	X	
Server Extensions		X
Server Management	X	X
System Management Hub	X	X
Windows Viewer	X	X

A shortcut for the Windows Viewer is always created. However, it is only functional in a production environment, if you have a license for the GUI Version.

Note:
Shortcuts for the Web Viewer, Terminal Viewer (browser version) and XML Version are not available since the HTML pages for these viewers have to be customized individually.

Program Folders

By default, Entire Screen Builder (development and production versions) is installed in the following folder:

Program Files\Software AG\Entire Screen Builder 5 - see the table below for further information.

The following exceptions apply for the standalone versions:

- Only one folder is created for the standalone version of the Terminal Viewer:

  Program Files\Software AG\Entire Screen Builder 5\Terminal Viewer Standalone

- Only one folder is created for the standalone version of the Windows Viewer:

  Program Files\Software AG\Entire Screen Builder 5\Windows Viewer

The following folders are created for the development and production versions:

Program Folder	**Contents**
..\Entire Screen Builder 5	*Readme.txt*
..\Entire Screen Builder 5\bin	EXE and DLL files.
..\Entire Screen Builder 5\data	Configuration file *nswconfig.xml* with information on the defined host sessions (including key schemes and color schemes) and the *profiles* folder with information on the defined user and group profiles. All of this information is defined with the System Management Hub.
..\Entire Screen Builder 5\doc	Full documentation set in HTML format, including PDF. Can be accessed from the Windows Start menu.
..\Entire Screen Builder 5\help	Help file *nswnnnUS.chm*, where *nnn* is the version number. Can be accessed from Entire Screen Builder's **Help** menu.
..\Entire Screen Builder 5\mmc	Files for the Server Management snap-in.
..\Entire Screen Builder 5\repository	Local repository folders and files. Empty after installation.
..\Entire Screen Builder 5\samples	Several subfolders containing a demo application and samples for data transfer, script files and user exits.
..\Entire Screen Builder 5\scripts	Script files. The two subfolders for production and test purposes are empty after installation.

Program Folder	Contents
..*Entire Screen Builder 5\server extensions*	This folder is only created when the Microsoft Internet Information Server (IIS) is installed. It contains the following subfolders: • *esbscripts* with DLLS for XML and tunneling. • *xml* where the subfolder *esb* contains *esblogon.htm* and directory images. The *xsl* subfolder contains further subfolders: *Netscape4*, *Netscape7* and *NoFrame*. • *tunneling* with *Tunneling.reg*.
..*Entire Screen Builder 5\tables*	Translation tables.
..*Entire Screen Builder 5\table_backup*	In the case of an update installation, your old *tables* folder is copied to *table_backup*. If you still want to use your old customized tables, you have to copy them to the new *tables* folder.
..*Entire Screen Builder 5\temp*	For internal use. Empty after installation.
..*Entire Screen Builder 5\terminal viewer*	*ExampleAdministratorTerminalPage.htm*, *NSWTerminalViewer.cab* and *ExampleTerminalViewer.htm*.
..*Entire Screen Builder 5\traces*	Traces. Empty after installation.
..*Entire Screen Builder 5\web viewer*	*NWWClientFull.cab*, *ExampleAdministratorPage.htm* and *ExampleEndUserPage.htm*. *NWWClientFull-unsigned.cab* can be used if you want a signing different from VeriSign.
..*Entire Screen Builder 5\windows viewer*	*EWVViewer.exe* and DLL files for the Windows Viewer.
..*Entire Screen Builder 5\xml*	XML templates for the System Management Hub.
..*Entire Screen Builder 5\xmlrepository*	Files for the XML Version. Empty after installation.

Environment Variables

Entire Screen Builder does not change any environment variables.

Registry

In the Windows registry, the Entire Screen Builder installation procedure sets up keys in:

- **HKEY_LOCAL_MACHINE/SOFTWARE/Software AG/Entire Screen Builder**
  When you uninstall Entire Screen Builder, these keys are automatically removed.
- **HKEY_CURRENT_USER/Software/Software AG/Entire Screen Builder**
  Entire Screen Builder uses these keys to store runtime information (e.g. last scope selected). You can delete these keys after uninstalling Entire Screen Builder.

For the Server Management tool, which is a Microsoft Management Console snap-in, several keys are added in other branches of the registry.

Using the Demo Application

The demo application can be invoked from the mainframe, UNIX and OpenVMS version of Natural. The following program folders of Entire Screen Builder pertain to the demo application:

Program Folder	Contents
..\Entire Screen Builder 5\samples\sampleconf	Rules folder for the demo application.
..\Entire Screen Builder 5\samples\sampleconf\natlib	Natural objects for the mainframe version of Natural, including the files *Natdemo.ncd*, *Employees.ncm* and *Vehicles.ncm* (see below).

The demo application is based on the Vehicles and Employees files. Therefore, the file IDs of these files must be known. To use the demo application, proceed as described below.

The demo application requires that the Natural message line is located at the bottom of the screen. If it is located at another position, use the Natural command `%MB` before starting the demo application.

Note:
For UNIX and OpenVMS:
The demo application is installed during the installation of the UNIX and OpenVMS components. See the following sections for further information: *Installing Natural for Entire Screen Builder on UNIX Hosts* and *Installing Natural for Entire Screen Builder on OpenVMS Hosts*.

To upload the demo application (only required for the mainframe version of Natural)

- Upload the file *Natdemo.ncd* using Natural's NATLOAD utility (see the Natural documentation for further information).

  This transfers the demo application to the Natural library NSWDEMO.

To upload the data definition modules (only required for the mainframe version of Natural)

1. In the Natural command line, enter "NTCPC".

   This displays the main menu of the NTCPC utility.
2. In the Code field, enter "D" and press ENTER.

   This displays the Data Definition Module screen.
3. In the Code field, enter "U".
4. In the DDM Name field, enter "employees".
5. Make sure that the field Accept existing DBID/FNR is set to "N".
6. Press ENTER.

   An upload window appears.
7. Change to the folder ..*Entire Screen Builder 5**samples**sampleconf* *natlib*.
8. Select the file *Employees.ncm*.
9. Choose the **Upload** button.

   A Natural window is now shown, displaying default values for DBID and FNR.
10. Change the values for DBID and FNR according to your requirements and press ENTER.
11. Repeat the above steps to upload the file *Vehicles.ncm* (enter "vehicles" in the DDM Name field).

To define a session for host access

1. Start the System Management Hub.
2. Define a session which allows you to invoke Natural.

   See *Host Sessions* in Entire Screen Builder's *System Management Hub* documentation for further information.

To define the rules repository

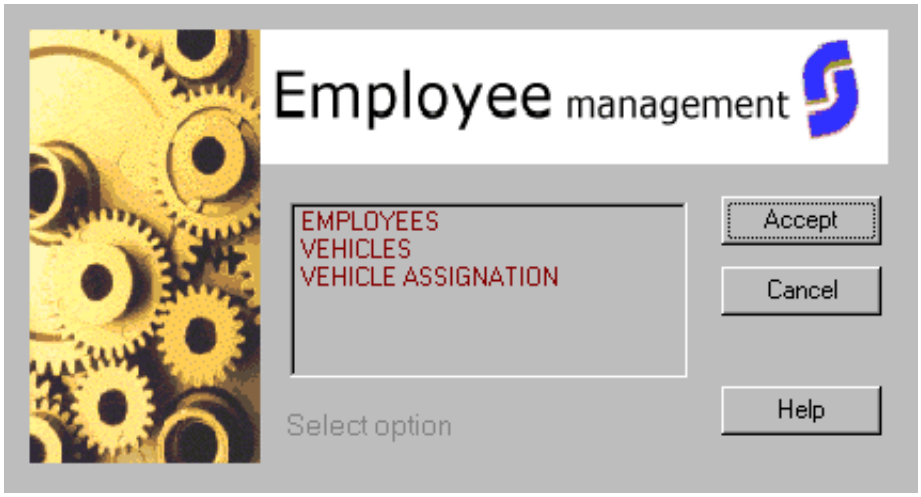
1. Start the Entire Screen Builder SDK.
2. From the **Options** menu, choose **Configuration**.
3. In the resulting Configuration dialog box, set the rules folder to ..*Entire Screen Builder 5**samples**sampleconf*.
4. Choose the **OK** button.
5. Quit the Entire Screen Builder SDK.
6. In the System Management Hub, stop and start the Entire Screen Builder Server. See *Starting and Stopping the Entire Screen Builder Server* in Entire Screen Builder's *System Management Hub* documentation.

To invoke the demo application

1. Start the Entire Screen Builder SDK.
2. Connect to the session you have defined.

   The session appears in the Windows Viewer.
3. Navigate to Natural.
4. Logon to the library NSWDEMO.
5. Start the demo application by entering "menu" on the NEXT prompt.

   The main menu should now look as follows:



Exporting Objects from Entire Connection to Entire Screen Builder

Starting with Entire Connection Version 4.3.1, Entire Connection's Configuration Manager provides an **Export** command. Using this command, you can transfer session settings, color schemes and key schemes from your Entire Connection share file to Entire Screen Builder. See your Entire Connection documentation for detailed information.

Installing the Entire Screen Builder Server in the UNIX Environment

You only have to install the Entire Screen Builder Server in a UNIX environment if you do not want to install it under Windows.

In order to install Entire Screen Builder successfully, you must have superuser (root) rights.

The setup program checks for existing Entire Screen Builder Version 4.3.1 and 5.1.1 installations. If such an installation is detected, the setup program offers the option to update to the new version and remove the existing version. It is recommended that you remove the existing version. To do so, you have to stop the Entire Screen Builder Server; otherwise, the existing version cannot be removed. Before starting the installation, it is therefore recommended that you inform all connected users (for example, with the Send Message function in the System Management Hub) and close all Entire Screen Builder viewers.

This chapter covers the following topics:

- Differences between the Windows and UNIX Versions
- Prerequisites
- License Files
- Setting Up the Entire Screen Builder Server
- Starting and Stopping the Entire Screen Builder Daemon
- Activating Entire Screen Builder at System Startup
- Entire Screen Builder Server Directories
- Working with the Entire Screen Builder Server in the UNIX Environment
- Deinstalling the Entire Screen Builder Server

See also: *Updating Manually Deployed Files on the HTTP Server*.

Differences between the Windows and UNIX Versions

The following table lists the differences between the UNIX and Windows versions of the Entire Screen Builder Server.

	Windows	**UNIX**
Server configuration information	Stored in the Windows registry.	Stored in the file *ewvreg.reg*. The environment variable WINREG_FILE points to this file.
Event log	Stored in the Windows Event Viewer.	Stored in the file *ewvserver.txt* which is located in the directory *$EWVDIR/$EWVNODE/events*.

Prerequisites

Server	Sun Solaris Version 8 (64 bit) or 9 (64 bit).
System Management Hub	Software AG's System Management Hub Version 3.3.1.2 or above. For supported browsers, see the System Management Hub documentation.
Tunneling	For tunneling, you can use the following HTTP server: • Apache Web Server Version 1.3 and 2.0 for Solaris (32 bit and 64 bit). See *HTTP Tunneling* for further information. If tunneling is not used, you can use any HTTP server.

The prerequisites for the presentation clients are listed in the section *Prerequisites for Windows*.

Note:
See also: *readme-ux.txt* in the *\unix\sun\ewv\v5nn* directory of the Entire Screen Builder CD.

Operating System Settings

The default value for "Maximum number of file descriptors" on Solaris 8 and 9 systems is 256. You may have to adjust this limit for your installation.

The Entire Screen Builder Server needs at least 2 file descriptors per connection (one socket for the connection to the viewer and another one for the connection to the host). It may need more file descriptors in specific situations. You can use the following rule:

```
MaximumNumberOfFilesDescriptors = (MaximumNumberOfClientsExpected * 2)+100
```

If tunneling is used, the Entire Screen Builder Server needs at least 3 file descriptors per connection. Use the following rule then:

```
MaximumNumberOfFilesDescriptors = (MaximumNumberOfClientsExpected * 3)+100
```

Please ask your system administrator to change the maximum number of file descriptors in your system.

License Files

During the installation, you will be asked to provide a license key. You should have a license key that is provided by Software AG either on a storage medium such as a diskette in the Entire Screen Builder distribution kit or by email.

Depending on the license file you specify during installation, one of the following is installed:

License File	Description	Type of Environment
EWV*nnn*.xml	Entire Screen Builder GUI Version.	Production environment.
ETV*nnn*.xml	Entire Screen Builder Terminal Version.	Production environment.
EMV*nnn*.xml	Entire Screen Builder XML Version.	Production environment.

Note:
The Entire Screen Builder Server can only be installed as a production environment version under UNIX. The development environment version (SDK) is only available for Windows platforms.

If you want to install all versions, you have to install one version first. Then start the installation once more. You can then select the license file for the desired product. As a result, the product is installed and the directories for the corresponding viewers are created.

Setting Up the Entire Screen Builder Server

Setting up the Entire Screen Builder Server on UNIX consists of the following steps:

- Step 1: Stop the Entire Screen Builder Server
- Step 2: Establish the Environment and Install the Product

Step 1: Stop the Entire Screen Builder Server

This step is only required, if there is a previous version of the Entire Screen Builder Server under UNIX. In this case, you have to stop the Entire Screen Builder Server and the System Management Hub as described below.

Stop the Entire Screen Builder Server in one of the following ways:

- Recommended. Stop the Entire Screen Builder Server from the System Management Hub. See *Starting and Stopping the Entire Screen Builder Server* in Entire Screen Builder's *System Management Hub* documentation.
- Invoke the shell script *ewvserver.sh* from the command line:

  ```
  ewvserver.sh stop
  ```

Stop the System Management Hub after the Entire Screen Builder Server has been stopped:

- Use the script *argsrvs.bsh* which is located in *../common/arg/bin*. This script can be used for both starting and stopping the System Management Hub.

Step 2: Establish the Environment and Install the Product

Read the file *setup_ux.txt* in the root directory of the Entire Screen Builder CD. Follow the instructions in this file in order to establish the environment for installing Software AG products and to install the product. The Entire Screen Builder-specific settings are shown below:

EWVDIR	Home directory for the product.
EWVVERS	Current version of the product.
EWVNODE	Name of the node on which Entire Screen Builder is installed.
WINREG_FILE	Contains the configuration file used by the server.
LD_LIBRARY_PATH	Must point to *$EWVDIR/$EWVVERS/bin*.

Complete the environment settings as specified in the file *sagenv.new* which is created during installation in the root directory of the Software AG products. When you have set the environment variables for Entire Screen Builder, you either have to start the System Management Hub daemons or, if the System Management Hub was already installed, you have to restart them. Use the script *argsrvs.bsh* from */common/arg/bin* to do this.

Verify the correct installation of the application as follows:

- Invoke the System Management Hub and create a new session (for example, Telnet TN3270). See *Host Sessions* in Entire Screen Builder's *System Management Hub* documentation.
- Start the Entire Screen Builder Server from the System Management Hub. See *Starting and Stopping the Entire Screen Builder Server* in Entire Screen Builder's *System Management Hub* documentation.
- From a Windows PC, open the HTML page *http://UnixServer/ESB/webviewer/ExampleAdministratorPage.htm* with the Microsoft Internet Explorer to start the Entire Screen Builder Web Viewer.
- When the Entire Screen Builder Web Viewer is shown in the Internet Explorer, specify all required information and choose the **Connect** button. For a default installation, it is sufficient when you specify the name of the host on which the Entire Screen Builder Server has been installed in the **ESB Server** text box.
- If you want to use the Server Management, you also have to start the Entire Screen Builder daemon. See *Starting and Stopping the Entire Screen Builder Daemon*.

Starting and Stopping the Entire Screen Builder Daemon

The Entire Screen Builder daemon is responsible for accepting the commands from the Server Management (which is a Microsoft Management Console snap-in).

When you start the Entire Screen Builder daemon for the first time, you have to change the owner and permission for the daemon as follows:

1. Change to *$EWVDIR/$EWVERS/INSTALL*.
2. Change to superuser (su or sudo).
3. Execute the script *ewvsudo.bsh*.

To start the Entire Screen Builder daemon:

```
ewvdaemon.sh nodename port-number start
```

Example using the default port number:

```
ewvdaemon.sh sunhost 22390 start
```

To stop the Entire Screen Builder daemon:

```
ewvdaemon.sh nodename port-number stop
```

Example:

```
ewvdaemon.sh sunhost 22390 stop
```

Please note that the same port number used to start the Entire Screen Builder daemon must be configured in the Server Management snap-in (the name of the corresponding option is **The computer is a UNIX system with a daemon port**. See *Defining the Entire Screen Builder Server* in the *Server Management* documentation for further information.

Activating Entire Screen Builder at System Startup

This section describes how to configure the UNIX system for starting Entire Screen Builder automatically during system startup.

The following table shows where the *init.d* and *rc3.d* directories are located on Solaris systems. In the following descriptions, *init.d* or *rc3.d* stand for the relevant path indicated below.

System Directory for Initialization	Run Level Startup Directory
/etc/init.d	*/etc/rc3.d*

The *rc3.d* directory contains several Bourne shell scripts that start with "S" and "K", followed by a number, for example "99". If you add a file to this directory, the code contained in it is executed when the system changes to "multi-user mode". The Bourne shell scripts that start with "S" are called by the UNIX system when the system is going up. The Bourne shell scripts that start with "K" are called by the UNIX system when the system is going down.

The following topics are covered below:

- Preparing the System V Style Startup Procedures for Entire Screen Builder
- Changing the System Management Hub Startup Procedure for Entire Screen Builder

Preparing the System V Style Startup Procedures for Entire Screen Builder

Entire Screen Builder provides two procedures for starting its servers at system startup. These are automatically unpacked to the *$EWVDIR/$EWVVERS/INSTALL* directory during the Entire Screen Builder installation process.

Procedure	Description
sagewvsrv	Used to invoke the Entire Screen Builder Server during system startup.
sagewvdmn	Used to invoke the Entire Screen Builder daemon during system startup.

Samples of the procedure files are shown below. They can be edited with a text editor.

Log in as user "root".

Copy the above two procedures to the *init.d* directory of your UNIX machine.

Check the environment variable settings (SAG, EWVVERS, EWVADM) in the two procedures.

EWVDIR	Location where Entire Screen Builder has been installed.
EWVVERS	Entire Screen Builder version number.
EWVADM	The login name of the Entire Screen Builder system administrator. It is assumed that this administrator account is called "sag" and that the user ID is already known to the system. It does not have to be a user with root privileges.

Note:
The Bourne shell does not allow blanks before and after the equal sign in the lines to be customized.

Go to the *rc3.d* directory.

Create backup copies of the current procedure files *S99ewvsrv* and *S99ewvdmm* in the *rc3.d* directory.

Copy the files starting with "sag" (for example, *sagewvsrv*) from the *init.d* directory to the *rc3.d* directory and rename them so that they start with "S99" (for example, *S99ewvsrv*).

Copy the files starting with "sag" from the *init.d* directory to the *rc3.d* directory once more and rename them so that they start with "K99" (for example, *K99ewvsrv*).

If an error occurs, you can start again with the backup copies after the deletion of the modified "S99" and "K99" files contained in the *rc3.d* directory.

Sample of the *sagewvsrv* File

```
#!/bin/sh
#
# Copyright (c) 2002 Software AG, Germany. All rights reserved.
#
# Start/stop script for Entire Screen Builder server
#
#====================================================================
EWVDIR="/usr/SAG/ewv"
EWVVERS="v521"
EWVADM="sag"
. $SAG/ewv/$EWVVERS/INSTALL/ewvenv
ewvserver=$EWVDIR/$EWVVERS/bin/ewvserver.sh
#
#--------------------------------------------------------------------
#
case "$1" in
  start)
    echo "Starting Entire Screen Builder server ..."
    if [ -x "${ewvserver}" ] ; then
      su $EWVADM -c "${ewvserver} start" > /dev/console 2> /dev/console
    else
      exit 1
    fi
    echo "done..."
    ;;
  stop)
    echo "Stopping  Entire Screen Builder server ..."
    if [ -x "${ewvserver}" ] ; then
      su $EWVADM -c "${ewvserver} stop" > /dev/console 2> /dev/console
    else
      exit 1
    fi
    ;;
  *)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac
#
#====================================================================
#
```

Sample of the *sagewvdmm* File

```
#!/bin/sh
#
# Copyright (c) 2002 Software AG, Germany. All rights reserved.
#
# Start/stop script for Entire Screen Builder daemon
#
#=================================================================
EWVDIR="/usr/SAG/ewv"
EWVVERS="v521"
EWVADM="sag"
. $EWVDIR/$EWVVERS/INSTALL/ewvenv
ewvdaemon=$EWVDIR/$EWVVERS/bin/ewvdaemon.sh
#
#-----------------------------------------------------------------
#
case "$1" in
  start)
    echo "Starting Entire Screen Builder daemon ..."
    if [ -x "${ewvdaemon}" ] ; then
      su $EWVADM -c "${ewvdaemon} start" > /dev/console 2> /dev/console
    else
      exit 1
    fi
    echo "done..."
    ;;
  stop)
    echo "Stopping  Entire Screen Builder daemon ..."
    if [ -x "${ewvdaemon}" ] ; then
      su $EWVADM -c "${ewvdaemon} stop" > /dev/console 2> /dev/console
    else
      exit 1
    fi
    ;;
  *)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac

#
#=================================================================
#
```

Changing the System Management Hub Startup Procedure for Entire Screen Builder

The procedure *sagarg* which is used to invoke the System Management Hub during system startup must be changed to set the Entire Screen Builder environment before the System Management Hub is started.

A sample of the procedure file is shown below. It can be edited with a text editor.

Log in as user "root".

Create a backup copy of your current *sagarg* file contained in the *init.d* directory.

Add the lines to the *sagarg* file which are indicated in bold in the sample file below.

Note:
The Bourne shell does not allow blanks before and after the equal sign in the lines to be customized.

If an error occurs, you can start again with the backup copy after the deletion of the modified *sagarg* file contained in the *init.d* directory.

Sample of the *sagarg* File

You have to add the lines indicated in bold (where */usr/SAG* is the path to the directory in which Entire Screen Builder has been installed).

```
#!/bin/sh
#
# Copyright (c) 2003 Software AG, Germany. All rights reserved.
#
# Start/stop script for System Management Hub - Server
#

SAG="/usr/SAG"
export SAG
. "/usr/SAG/common/arg/INSTALL/argenv" > /dev/null
. "/usr/SAG/ewv/v521/INSTALL/ewvenv" > /dev/null

SAGUSER="sag"
PIDFILE="$ARGDIR/var/argsrv.lock"

case "$1" in
  start|resume)
    echo "Starting Software AG System Management Hub - Server..."
    # First check if the pidfile had become stale...
    if [ -r "$PIDFILE" ] ; then
        ps -fp `cat "$PIDFILE"` | tail "+2" > /dev/null 2>&1 || {
            rm $PIDFILE
        }
    fi
    if [ -r "$PIDFILE" ] ; then
        echo "argsrv still running ... not started."
    else
        if [ -x "$ARGDIR"/bin/argsrv ]; then
            su $SAGUSER -c "/bin/sh -c "'"'" \
                . \"$SAG\"/common/arg/INSTALL/argenv > /dev/null; \
                . /usr/SAG/ewv/v521/INSTALL/ewvenv > /dev/null; \
                cd \"$ARGDIR\"/bin; ./argsrv -pid \"$PIDFILE\""'"'" "
        else
            exit 1
        fi
    fi
    echo "done..."
    ;;
  stop|pause)
    echo "Stopping Software AG System Management Hub - Server..."
    case "$1" in pause) SIG=3 ;; *) SIG=15 ;; esac
    kill -$SIG `cat "$PIDFILE"`; rm -f "$PIDFILE"
    echo "done..."
    ;;
  *)
    echo "Usage: $0 {start|resume|stop|pause}"
    exit 1
    ;;
esac
```

Entire Screen Builder Server Directories

The following directories are created when Entire Screen Builder is installed in the UNIX environment:

Directory	Description
ewv	Top-level directory.
v5xxx	Directory with all components for the current Entire Screen Builder version.
v5xxx/INSTALL	Shell scripts and environment files to install the product.
v5xxx/bin	Entire Screen Builder executable and library files.
v5xxx/conf-templates	Template files (*ewvreg.reg* and *ewvargus.reg*).
v5xxx/data	Configuration file *nswconfig.xml*.
v5xxx/doc	Online documentation.
v5xxx/images	Images for the System Management Hub.
v5xxx/samples	Miscellaneous samples for user exits and scripts.
v5xxx/samples/sampleconf	Demo application and samples.
v5xxx/samples/samplenattransfer	Sample programs for data transfer with Natural.
v5xxx/samples/samplescripts	Sample Java Scripts.
v5xxx/samples/sampleuserexit	Sample user exits.
v5xxx/tables	Translation tables.
v5xxx/terminalviewer	Only installed for the Entire Screen Builder Terminal Version. *NSWTerminalViewer.cab* and *ExampleTerminalViewer.htm*.
v5xxx/webviewer	Only installed for the Entire Screen Builder GUI Version. *NWWClientFull.cab*, *ExampleAdministratorPage.htm* and *ExampleEndUserPage.htm*.
v5xxx/windowsviewer	Only installed for the Entire Screen Builder GUI Version. *EwvViewer.exe*.
v5xxx/xml	XML templates for the System Management Hub.
systemname	Directory with the specific configuration for this node.
systemname/configuration	Configuration file *ewvreg.reg*.
systemname/data	Configuration file *nswconfig.xml* with information on the defined host sessions (including key schemes and color schemes) and the *profiles* folder with information on the defined user and group profiles. All of this information is defined with the System Management Hub.
systemname/events	The file *ewvserver.txt* with information on each event that occurred on the server.
systemname/logs	Log files.

Directory	Description
systemname/repository	Rules repository.
systemname/scripts	Script files. The two subfolders for production and test purposes are empty after installation.
systemname/tables	Translation tables for the current system.
systemname/temp	For internal use. Empty after installation.
systemname/traces	Traces files when the server is running in trace mode.
systemname/xmlrepository	Files for the XML Version. Empty after installation.

systemname is the name of the machine. This directory is created automatically during the installation. The folders and files in *systemname* are the working folders and files used by the Entire Screen Builder Server and the administration tool at runtime.

Working with the Entire Screen Builder Server in the UNIX Environment

In the UNIX environment, the Entire Screen Builder Server works only in production mode. This means that the rules have to be created with the Entire Screen Builder SDK under Windows.

Once the rules have been created, you must copy the entire repository (*defaults.ini*, **.bdd*, images etc., including all subdirectories) to the directory *$EWVDIR/$EWVNODE/repository* in the UNIX file system.

Since UNIX distinguishes between lowercase and uppercase, each file name in the repository must be in lowercase. To convert all file names to lowercase, change to the directory *$EWVDIR/$EWVNODE/repository* and run the shell script *up2lofn.sh*.

To use the new rules, you have to stop the Entire Screen Builder Server and then start it again. See *Starting and Stopping the Entire Screen Builder Server* in Entire Screen Builder's *System Management Hub* documentation.

Deinstalling the Entire Screen Builder Server

To deinstall the Entire Screen Builder Server in the UNIX environment, go to the $SAG environment and run `sagrm`. Select all items to be deinstalled in the resulting screen.

Updating Manually Deployed Files on the HTTP Server

After having updated the Entire Screen Builder Server with the setup program (either on Windows or UNIX), you have to manually update all files and modules on the HTTP server.

1. Update the HTML pages for the viewers you are using.

   If you created your own HTML pages, you have to update the version number in the codebase attribute to the current version number, including the current patch level. Example for the Web Viewer:

   ```
   codebase="http://www.myserver/NSW/NWWClientFull.cab#Version=5,2,1,0"
   ```

2. Copy the new CAB files for the viewers to the HTTP server.

3. If HTTP tunneling is used, update the appropriate tunneling modules with the new files from the CD.

   To copy the files, you have to stop the HTTP server first. For example, to update the ISAPI DLL for your Microsoft Internet Information Server, you must first stop the service using the IIS management tool.

   See *HTTP Tunneling* for further information.

Installing Natural for Entire Screen Builder on UNIX Hosts

If the host system on which you run your Natural applications is a UNIX system, additional software for Entire Screen Builder has to be installed on the host. The Entire Screen Builder UNIX modules are shipped on the Natural UNIX CD.

In general, Entire Screen Builder uses the default system parameter values provided with the UNIX system.

This chapter covers the following topics:

- Prerequisites
- Setting Up the Entire Screen Builder Components
- Directories
- Configuration Files
- Working with the Entire Screen Builder UNIX Components

Prerequisites

Supported Operating Systems	The same platforms as supported by Natural UNIX.
Other Software Products	Natural Version 5.1.1.9 or above.
Linker	A linker (for example, ld or cc) and the command `make` must be available in the system.

Setting Up the Entire Screen Builder Components

Setting up Entire Screen Builder on UNIX consists of the following steps:

- Step 1: Stop the Entire Screen Builder Daemons
- Step 2: Establish the Environment
- Step 3: Install Natural and Entire Screen Builder with the Demo Application
- Step 4: Check the Environment Variables for Entire Screen Builder
- Step 5: Read the READ_NSW Files

Step 1: Stop the Entire Screen Builder Daemons

This step is only required for an upgrade installation. It is not required when you install Entire Screen Builder for the first time.

Stop the *nswsrvd* process using the following command:

```
nswsrvd.sh servicename stop
```

Repeat this command for each Entire Screen Builder service that has been started.

Step 2: Establish the Environment

Ensure that the environment definitions, as described in *setup.txt* in the root directory of the Natural CD, are correct and set.

Step 3: Install Natural and Entire Screen Builder with the Demo Application

Entire Screen Builder and the demo application are installed during the Natural installation.

Important:
The Natural installation provides an Entire Screen Builder option which must be activated. For more details, see your Natural installation documentation.

When you install Natural and Entire Screen Builder, the directory *$NATDIR/$NSWNODE* is created. The template files located in *$NATDIR/$NATVERS/nsw/node-name* are then copied to this new directory.

Once Natural and Entire Screen Builder with the demo application are installed, you must catalog the application SYSEXNSW by entering the following commands:

```
natural
logon SYSEXNSW
catall
```

Step 4: Check the Environment Variables for Entire Screen Builder

The Entire Screen Builder-specific settings are shown below:

Environment Variable	Description
NSWDIR	Home directory for the product.
NSWNODE	Name of the node on which Entire Screen Builder is installed.
NSWSERV	Name of the path to the *nswservice* file.
NSWTIMEOUT	Number of seconds waiting for an answer from the PC side.

Step 5: Read the READ_NSW Files

Access the directory *$NATDIR/$NATVERS* and read the *READ_NSW.1ST* and *READ_NSW.FIX* files for any version-specific installation considerations concerning the particular platform.

Add the services as described in the file *READ_NSW.1ST*.

Directories

The following directories are created when Natural is installed together with Entire Screen Builder on a UNIX system:

Directory	Description
$NATDIR	Top-level Natural directory.
$NATDIR/$NATVERS	Directory with all components for the current Natural version.
$NATDIR/$NATVERS/nsw	Directory with some Entire Screen Builder components for the current version.
$NATDIR/$NATVERS/INSTALL	Shell scripts and environment files to install the product.
$NATDIR/$NATVERS/nsw/bin	Entire Screen Builder executable files (*nswusr*, *nswsrvd*, *nswusr.tr* and *nswsrvd.tr*).
$NATDIR/$NATVERS/nsw/node-name	Contains the template files (*services.dat*, *nswservice*, etc.).
$NATDIR/$NATVERS/nsw/samples/userexit	Contains the files for building the sample user exit.
$NATDIR/$NATVERS/bin/build	Contains the library (*libnsw.a*) to link with Natural.
$NATDIR/$NATVERS/bin/build.tr	Contains the trace library (*libnsw.a*) to link a trace version with Natural.
$NATDIR/$NSWNODE	Contains the configuration files (*services.dat*, *nswservice*, etc.).

Note:
The above table lists only the most important directories and files.

Configuration Files

When the Entire Screen Builder installation has finished, the directory *$NATDIR/$NSWNODE* contains the following configuration files:

Configuration File	Description
nsw.sh	Shell script to start the Natural application.
nswservice	Contains the authorized users for every service.
nswsrvd.sh	Shell script to start and to stop the NSW daemon.
services.dat	Contains the configuration for the NSW daemon.

Working with the Entire Screen Builder UNIX Components

The Entire Screen Builder UNIX components are used to start the Natural applications linked with the Entire Screen Builder library.

The following topics are covered below:

- Starting a New Natural Application
- Starting and Stopping the Entire Screen Builder Daemon

Starting a New Natural Application

Any Natural application can be used with Entire Screen Builder.

To start a new Natural application with Entire Screen Builder, proceed as follows:

- Create a new parameter file using the Natural Parameter Utility (see the Natural documentation) and modify the STACK command as follows:

  ```
  logon library; startprogram; fin
  ```

Add the new service as follows:

1. Insert a new line in the file */etc/services*:

   ```
   servicename portnumber/tcp # Comment
   ```

2. Create a new shell script (similar to *nsw.sh*) to startup the Natural application:

   ```
   $NATDIR/$NATVERS/bin/natnsw parm=new-parameter-file etid=$$ >/dev/null 2>&1 &
   ```

3. Optional. If function keys and message lines are to be displayed in their native format (i.e. as normal text), set the environment variable NSW_PF_MSG_LINES_NATIVE_FORMAT to "YES". To do so, insert the following two lines before the natnsw line (see the previous step):

   ```
   NSW_PF_MSG_LINES_NATIVE_FORMAT="YES"
   export NSW_PF_MSG_LINES_NATIVE_FORMAT
   ```

   If NSW_PF_MSG_LINES_NATIVE_FORMAT is not set or if its value is not "YES", function keys and message lines are detected automatically (default). If they are to be treated in a special way, you have to define the basic rules Function Keys and Message Line in the same way as for a mainframe screen.

   This feature is available starting with Natural Version 4.1.2.21 and Natural Version 5.1.1.3.

4. Insert a new line in the file *$NATDIR/$NSWNODE/services.dat*:

   ```
   servicename user $NSWDIR/bin/nswusr security $NATDIR/$NSWNODE /shellscript
   ```

servicename	Services used as entered in the previous step. These service names are optional. You can use other names and more service names.
user	Owner of the processes that will be started (usually "sag").
nswusr	Entire Screen Builder application server.
security	Enter one of the following characters: A Security is enabled. *nswusr* checks whether user and password are correct and whether the password has expired. In addition, a warning message is shown when the password will expire in a few days. The user must be authorized in the *nswservice* file. C Security is enabled. User and password are checked by an external program. See *nsw_CheckUsernameAndPassword* in the *User Exits* documentation. D Security is disabled. User and password are not checked. U Security is enabled. *nswusr* checks whether user and password are correct and whether the password has expired. In addition, a warning message is shown when the password will expire in a few days.
shellscript	Name of your shell script for starting the Natural application. The provided *nsw.sh* is just a template that has to be modified to suit your specific needs.

5. Add the following entries to the authentication file *$NSWSERV*:

   ```
   servicename:user1, user2
   ```

   where *user1* and *user2* are authorized users. If several users are authorized, separate the users in the list with commas. These users must already be defined in the system.

   This service is now available for use with a PC. See *Using the Demo Application* for further information.

Starting and Stopping the Entire Screen Builder Daemon

The Entire Screen Builder daemons are responsible for accepting new sessions. These daemons can be started and stopped using the following command:

nswsrvd.sh *servicename* [start|stop]

Installing Natural for Entire Screen Builder on OpenVMS Hosts

If the host system on which you run your Natural applications is an OpenVMS system, additional software for Entire Screen Builder has to be installed on the host. The Entire Screen Builder OpenVMS modules are shipped on the Natural OpenVMS CD.

In general, Entire Screen Builder uses the default system parameter values provided with the OpenVMS system.

This chapter covers the following topics:

- Prerequisites
- Setting Up the Entire Screen Builder Components
- Directories
- Configuration File
- Setting Up and Activating the NSWSRVD Daemon

Prerequisites

Supported Operating Systems	OpenVMS Version 7.2 or 7.3.
Other Software Products	Natural Version 4.1.2.12 through 5.1.1.

Setting Up the Entire Screen Builder Components

Setting up Entire Screen Builder on OpenVMS consists of the following steps:

- Step 1: Stop the Entire Screen Builder Daemons
- Step 2: Establish the Environment
- Step 3: Install Natural and Entire Screen Builder with the Demo Application
- Step 4: Check the Environment for Entire Screen Builder
- Step 5: Define the TCP Port Number

Step 1: Stop the Entire Screen Builder Daemons

This step is only required for an upgrade installation. It is not required when you install Entire Screen Builder for the first time.

Stop the nswsrvd process using the following command:

```
stop servicename
```

Repeat this command for each Entire Screen Builder service that has been started.

Step 2: Establish the Environment

Ensure that the environment definitions, as described in *readme.txt* in the root directory of the Natural CD, are correct and set.

Step 3: Install Natural and Entire Screen Builder with the Demo Application

Entire Screen Builder and the demo application are automatically installed during the Natural installation.

When you install Natural and Entire Screen Builder, the directory *NATDIR:[nswnode]* is created, where *nswnode* contains the system name. The template files located in *NATDIR:[natvers.NSW]* are then copied to this new directory.

Once Natural and Entire Screen Builder with the demo application are installed, you must catalog the application SYSEXNSW by entering the following commands:

```
natural
logon SYSEXNSW
catall
```

Step 4: Check the Environment for Entire Screen Builder

Besides the logical names *NATDIR* and *natvers* as defined by Natural, Entire Screen Builder needs the following logical names which are created during the installation of Natural:

Logical Name	Description
nswnode	Contains the system name.
VAXC$PATH	Contains the physical device specification of *NATDIR:[natvers.BIN]*

Example:

```
Define VAXC$PATH="ALF9$DKB500:[NATURAL.V41212.BIN]"
```

In addition, the logical names *NATOW* and *NATFE* are redefined during the start of the daemon process to point to the Entire Screen Builder images *NATNSWnatvers.EXE* and *NATFENSWnatvers.EXE*.

Optional. If function keys and message lines are to be displayed in their native format (i.e. as normal text), set the environment variable `NSW_PF_MSG_LINES_NATIVE_FORMAT` to "YES":

```
Define NSW_PF_MSG_LINES_NATIVE_FORMAT="YES"
```

If `NSW_PF_MSG_LINES_NATIVE_FORMAT` is not set or if its value is not "YES", function keys and message lines are detected automatically (default). If they are to be treated in a special way, you have to define the basic rules Function Keys and Message Line in the same way as for a mainframe screen. This feature is available starting with Natural Version 4.1.2.21 and Natural Version 5.1.1.3.

Step 5: Define the TCP Port Number

The UCX service with the TCP port number must be defined in the system as follows:

```
$ UCX SET SERVICE NSWDEMO /PORT=22370 /FILE="" /USER="" /PROC=""
```

Instead of NSWDEMO and the above port number, you can also specify other values. For example, you can create or define the TCP service name NSWAPPL1 with the port number 25000.

Directories

The following directories are created when Natural is installed together with Entire Screen Builder on an OpenVMS system:

Directory	Description
NATDIR	Top-level Natural directory.
NATDIR:[natvers]	Directory with all components for the current Natural version.
NATDIR:[natvers.INSTALL]	Shell scripts and environment files to install the Natural product.
NATDIR:[natvers.BIN]	Entire Screen Builder executable files *NATFENSWnatvers.EXE*, *NATNSWnatvers.EXE* and *NSWSRVDnatvers.EXE*.
NATDIR:[natvers.FNAT]	Contains the Natural demo application SYSEXNSW for Entire Screen Builder.
NATDIR:[nswnode]	Contains the configuration files *NSWSRVD_servicename.COM*, *NSWSRVD_servicename.LOG*, *SERVICES.DAT* and *START_NSWSRVD.COM*.

The files *NSWSRVD_servicename.COM* and *NSWSRVD_servicename.LOG* are created when the Entire Screen Builder daemon *NSWSRVDnatvers.EXE* is started with the procedure *START_NSWSRVD.COM*.

servicename is the UCX service as defined in the file *SERVICES.DAT*.

natvers indicates the version number and patch level of the corresponding Natural version.

Configuration File

The configuration file *SERVICES.DAT* is located in the directory *NATDIR:[nswnode]*, where the *nswnode* contains the node name (for example, *NATDIR:[ALF9]SERVICES.DAT*).

The content of this configuration file is one line for each defined TCP service:

```
servicename username natural parm1 ... parmn
```

servicename	Must be the same name as used in the TCP port number definition (see above).
username	Not used.
natural	This is the program name which must not be changed.
parm1 … parmn	Dynamic Natural parameters.

Example:

```
nswdemo sag natural parm=mypar bp=bp1
nswapp1 sag natural parm=app1 bp=bp1
nswapp2 sag natural parm=app2 bp=bp2
```

Note:
If the *NSWSRVDnatvers* daemon does not detect Natural's dynamic parameter ETID, the daemon automatically adds the ETID to the list of the dynamic parameters to be passed to Natural. The ETID added by the daemon has the format `ETID=number_ username`. It is truncated if the string exceeds 8 characters.

Setting Up and Activating the NSWSRVD Daemon

The BYPASS privilege must be authorized for the account to start the Entire Screen Builder daemon. The BYPASS privilege must also be set for the daemon process.

When TCP port number and service have been defined (UCX) and the *SERVICES.DAT* template file has been modified according to your requirements, you can start the *NSWSRVD* daemon to use Entire Screen Builder.

To start the daemon, invoke the DCL procedure *START_NSWSRVD.COM* as follows:

```
@START_NSWSRVD.COM service natvers
```

service contains the name of the service as defined with UCX.

natvers defines the Natural version and patch level.

If both parameters *service* and *natvers* are omitted, the defaults NSWDEMO and the current Natural version are used. The command procedure creates the temporary file *NSWSRVD_ servicename.COM* which sets up the environment and creates all logicals for Entire Screen Builder and starts the daemon.

Once the daemon has been started, the file *NSWSRVD_ servicename.LOG* is created. This file contains information (including the errors) about the daemon.

NSWSRVD_ servicename.COM and *NSWSRVD_servicename.LOG* are located in the directory *NATDIR:[nswnode]*.

Note:
The account which starts the daemon must hold the privilege IMPERSONATE as the default privilege. It is not sufficient to have an authorized privilege.

Customizing the Web Viewer

The Entire Screen Builder Web Viewer runs in the browser of a client workstation. It runs in any browser which supports ActiveX controls (for example, Microsoft Internet Explorer Version 5).

This chapter covers the following topics:

- General Information
- HTML Page

General Information

After installation, the example HTML pages *ExampleAdministratorPage.htm* and *ExampleEndUserPage.htm* can be found in the Entire Screen Builder folder *web viewer*. These pages need to be customized by the administrator.

To make the Web Viewer available to all users in your environment, copy the file *NWWClientFull.cab* and your customized HTML pages to the HTTP server. No installation is required on a client workstation. Inform all users of the URL that is to be used to open the end-user HTML page.

Note:
NWWClientFull.cab is a signed CAB file that is certified by the VeriSign company. *NWWClientFull-unsigned.cab* can also be used if you want to use signing different from VeriSign.

When a browser supporting ActiveX controls opens the HTML page for the first time, it automatically downloads and installs the ActiveX control on the local PC. This control is downloaded and installed only once. Later when the same HTML page is opened again, the control is not downloaded again. The browser only downloads it once more, when the version number of the ActiveX control has changed.

Important:
Make sure that the security settings of your browser allow you to download and run ActiveX controls.

HTML Page

In addition to your script (e.g. JavaScript or VBScript), the HTML page must contain the following:

```
<object
id="cControl"
classid="CLSID:3BB4FE3B-7A37-11D3-A41E-0060080C03B3"
codebase="<URI>NWWClientFull.cab#Version=5,2,1,0"
align="baseline" border="0" width="800" height="600">
        <param name="BACKGROUND" value="192,192,192">
        <param name="EMBEDDED" value="1">
        <param name="SERVER" value="localhost">
        <param name="PORT" value="22367">
        <param name="CONNECTION" value="1">
        <param name="ANONYMOUS" value="1">
        <param name="REPOSITORY" value="ESB_Repository">
        <a name="control">Your browser doesn’t support ActiveX controls </a>
</object>
```

Id Attribute

You can specify any value for the `id` attribute. This is the object name which is to be used in your script in order to communicate with the ActiveX control or to invoke methods of the ActiveX control. Example:

```
<script language="VBScript">
<!--
Sub window_onLoad()
     cControl.connect()
end sub
-->
</script>
```

Classid Attribute

You must not change the value for the `classid` attribute. This value identifies the ActiveX class.

Codebase Attribute

Substitute `<URI>` with the URI (Universal Resource Indicator) to the file *NWWClientFull.cab*. For example:

```
codebase="http://www.myserver/NSW/NWWClientFull.cab#Version=5,2,1,0"
```

The *NWWClientFull.cab* file contains all files, including the ActiveX control, for running the Entire Screen Builder Web Viewer on the client side. This file is downloaded and decompressed for the browser when the HTML page is opened.

Param Tag

With the `param` tag, you can set the values for a property of the control. The default values are used for those properties that have not been specified.

A detailed description for these properties can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

See also: *HTTP Tunneling*.

Other Attributes

The attributes `align`, `border`, `width` and `height` define the object's appearance in the browser. In the above code, the object is shown without a border. Its width is 800 pixels and its height is 600 pixels.

You can use all attributes that are supported for the `OBJECT` element. See your HTML documentation for further information.

Customizing the Terminal Viewer

The information in this chapter only applies to the browser version of the Entire Screen Builder Terminal Viewer.

The browser version of the Terminal Viewer runs in the browser on a client workstation. It runs in any browser which supports ActiveX controls (for example, Microsoft Internet Explorer Version 5).

The following topics are covered:

- General Information
- HTML Page
- Making the Online Documentation Available to All Users

Note:
The standalone version of the Terminal Viewer does not require a customized HTML page.

General Information

After installation, the example HTML pages *ExampleAdministratorTerminalPage.htm* and *ExampleTerminalViewer.htm* can be found in the Entire Screen Builder folder *terminal viewer*. These pages need to be customized by the administrator.

To make the Terminal Viewer available to all users in your environment, copy the file *NSWTerminalViewer.cab* and your customized HTML page to the HTTP server. No installation is required on a client workstation. Inform all users of the URL that is to be used to open the HTML page.

Note:
The help file is not included in the file *NSWTerminalViewer.cab*. See *Making the Online Documentation Available to All Users*.

When a browser supporting ActiveX controls opens the HTML page for the first time, it automatically downloads and installs the Terminal Viewer ActiveX control on the local PC. This control is downloaded and installed only once. Later when the same HTML page is opened again, the control is not downloaded again. The browser only downloads it once more, when the version number of the ActiveX control has changed.

Important:
Make sure that the security settings of your browser allow you to download and run ActiveX controls.

HTML Page

In addition to your script (e.g. JavaScript or VBScript), the HTML page must contain the following:

```
<object
 id="cControl"
 classid="CLSID:48c4ba58-8555-4ef4-b2ad-b62770c1edec"
 codebase=<URI>NSWTerminalViewer.cab#Version=5,2,1,0"
 align="baseline" border="0" width="850" height="375">
 <param name="ServerName" value="localhost">
 <param name="PortNumber" value="22340">
 <param name="AnonymousLogon" value="1">
 <param name="Embedded" value="0">
 <a name="control">Your browser doesn’t support ActiveX controls </a>
</object>
```

Id Attribute

You can specify any value for the `id` attribute. This is the object name which is to be used in your script in order to communicate with the ActiveX control or to invoke methods of the ActiveX control. Example:

```
<input type="button" name="cmdConnect" value="Connect" language="VBScript"
 onclick="cControl.PortNumber = Document.cFormParameters.inputProteusPort.value
          cControl.ServerName = Document.cFormParameters.inputProteusServer.value
          cControl.AnonymousLogon = Document.cFormParameters.inputAnonymous.checked
          cControl.UsePCLogonName = Document.cFormParameters.inputUsePCUserName.checked
          call cControl.Initialise()">
```

Classid Attribute

You must not change the value for the `classid` attribute. This value identifies the ActiveX class.

Codebase Attribute

Substitute `<URI>` with the URI (Universal Resource Indicator) to the file *NSWTerminalViewer.cab*. For example:

```
codebase="http://www.myserver/NSW/NSWTerminalViewer.cab#Version=5,2,1,0"
```

The *NSWTerminalViewer.cab* file contains all files, including theActiveX control, for running the Entire Screen Builder Terminal Viewer on the client side. This file is downloaded and decompressed for the browser when the HTML page is opened.

Param Tag

With the `param` tag, you can set the values for a property of the control. The default values are used for those properties that have not been specified.

A detailed description for these properties can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

The Terminal Viewer can be run in embedded and non-embedded mode.

Tip:
It is recommended to run the Terminal Viewer in non-embedded mode. In embedded mode, you may experience noticeable screen flicker caused by the Microsoft Internet Explorer.

See also: *HTTP Tunneling*.

Other Attributes

The attributes `align`, `border`, `width` and `height` define the object's appearance in the browser. In the above code, the object is shown without border. Its width is 850 pixels and its height is 375 pixels.

You can use all attributes that are supported for the `OBJECT` element. See your HTML documentation for further information.

Making the Online Documentation Available to All Users

Because of its size, the help file *nswnnnUS.chm* is not included in the file *NSWTerminalViewer.cab*. If you want to make it available to all users, you have to create a registry key on each client workstation. Before you edit the registry, make sure you understand how to restore it if a problem occurs.

Warning:
Using the Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. We cannot guarantee that problems resulting from the incorrect use of the Registry Editor can be solved. Use the Registry Editor at your own risk.

Create the following registry key (string value) on each client workstation:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Software AG\Entire Screen Builder\Terminal Viewer\Documentation\HtmlHelpFile
```

For `HtmlHelpFile`, specify the path to the folder containing the file *nswnnnUS.chm*. For example:

```
C:\Program Files\Software AG\Entire Screen Builder 5\help\
```

It is recommended that this path points to a folder on a server. Otherwise, you have to copy the help file to each client workstation.

After editing the registry, start the browser version of the Terminal Viewer and make sure that the help file is invoked correctly: from the **Help** menu, choose **Online Documentation**.

Using the Viewers with Natural on UNIX and OpenVMS Hosts

This chapter covers the following topics:

- Logon Credentials
- Closing the Natural Application and Natural in Error Situations
- Restrictions

Logon Credentials

When logging on to UNIX and OpenVMS hosts, the following items control the way how the logon credentials are picked up:

Natural UNIX logon dialog

A logon dialog box can be shown to get the logon information for a UNIX or OpenVMS system. How to define this depends on the type of viewer.

- **Web Viewer**
  For the Web Viewer, the property `param name="UNIXLOGON" value="1"` is set in the HTML page. See *Customizing the Web Viewer*.
- **Windows Viewer**
  For the Windows Viewer, the property **Force Natural UNIX Logon** is set in the Client Control Properties dialog box. See *Defining the Client Control Properties* in the documentation *Defining the Rules Using the SDK*.
- **User Exits**
  For the Web Viewer and Windows Viewer, this can also be set with an Entire Screen Builder user exit. See the *User Exits* documentation for a description of the property `UnixLogon`.
- **Terminal Viewer**
  For the Terminal Viewer, the property **Prompt user for logon credentials** is set when defining host sessions of type Natural UNIX using the System Management Hub. See *Terminal Viewer Properties* in Entire Screen Builder's *System Management Hub* documentation.
- **JavaScript**
  For the Terminal Viewer, this can also be set using a method of the NSW object. See the *Script Files* documentation for a description of the method `SetUnixLogonCredentials`.

Use logon credentials above

This property is set when defining host sessions of type Natural UNIX using the System Management Hub. See *Communication Properties for Natural UNIX* in Entire Screen Builder's *System Management Hub* documentation.

The logon credentials (user ID and password) to be used are specified in the same dialog as the property **Use logon credentials above**.

Values from client

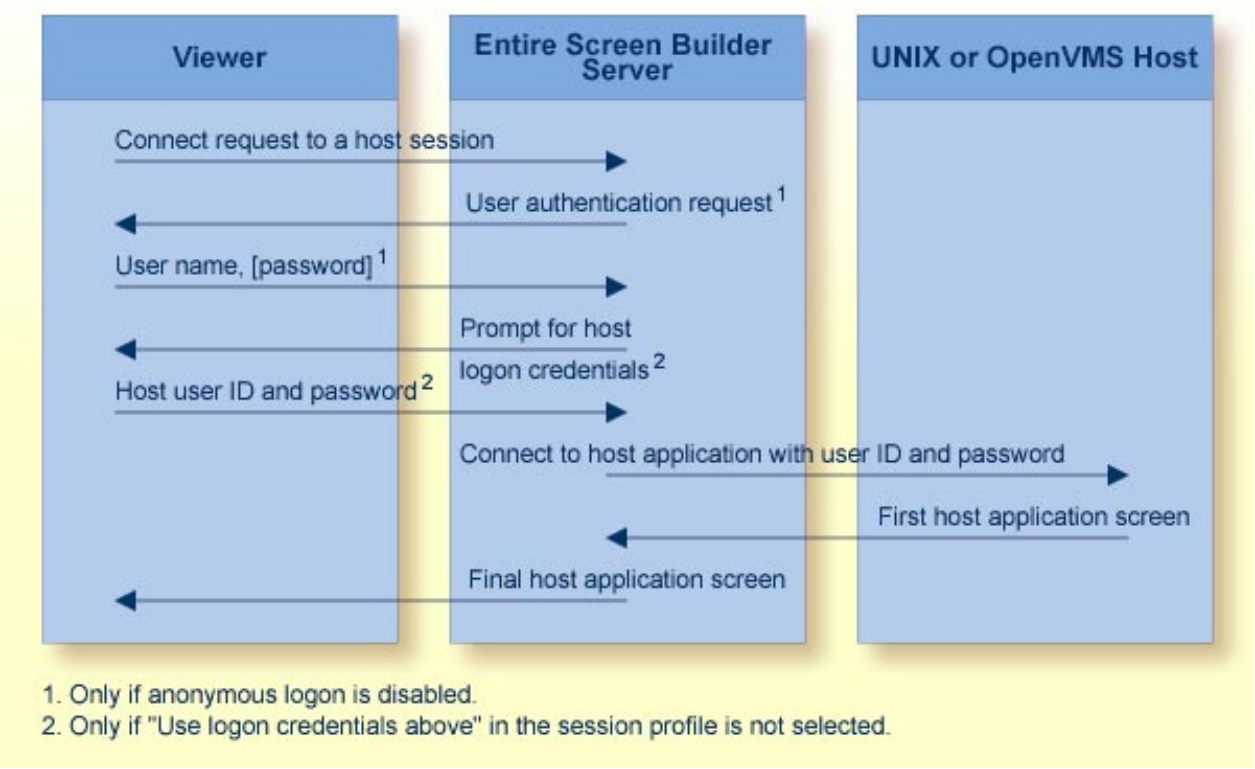
These are the logon credentials (user ID and password) that the user enters in the dialog box. The dialog box is only shown if Natural UNIX logon dialog has been enabled.

The following table shows the possible combinations of the above properties.

Natural UNIX logon dialog	Use logon credentials above	Values from client	Action
On	Off	Filled	Send values from client.
On	Off	Empty	Send empty string.
On	On	Filled	Send values from client.
On	On	Empty	Send values from session profile.
Off	On	Not applicable	Send values from session profile.
Off	Off	Not applicable	Send empty string.

Two different types of logon dialog box may appear: one for specifying the Natural UNIX and Natural OpenVMS logon credentials (see above) and, before that, another for specifying the user name as defined with the System Management Hub (when anonymous logon is not enabled); see *Users* in Entire Screen Builder's *System Management Hub* documentation.

The following graphic explains the user authentication process:



Closing the Natural Application and Natural in Error Situations

There are error situations in Entire Screen Builder which force the termination of the Natural session. For example, when the user chooses the close button in the upper right corner of the Entire Screen Builder window, the communication with the Natural application is disconnected immediately. Then also the Natural process finishes execution immediately without running through the normal close down code of the application. This may lead to inconsistencies in the system, for example, if the corresponding entry for the Natural process remains in the database user queue.

Entire Screen Builder returns error codes to Natural which can be used in a Natural error handling routine. To make sure that the close down code of your application is always executed, write an error handling routine which identifies the Entire Screen Builder fatal errors or add this code to your existing error handling routine. For critical errors, move the command `FIN` to the top of the Natural stack in order to finish and close everything correctly.

The Entire Screen Builder critical errors are:

6296	Fatal error in the communication. The communication will be disconnected immediately.
6297	Fatal error allocating memory.

Example: Application Program

This Natural program reads the employees file stored in Adabas. The program moves `ERRGEN` to `*ERROR-TA` to check any Natural error.

```
DEFINE DATA LOCAL
1 VIS VIEW OF EMP
   2 AA-1
END-DEFINE
MOVE "ERRGEN" TO *ERROR-TA
READ VIS
  DISPLAY AA-1
END-READ
END
```

Example: Error Handling Routine ERRGEN

This Natural program with the name ERRGEN checks the Natural error number. If the error number is 6296 (fatal error in the communication), the program stacks the command FIN on top of the Natural stack. The Natural execution finishes immediately and all databases are closed.

```
DEFINE DATA LOCAL
1 ERRNUMBER (N4)
1 LINENUMBER (N4)
1 STATUS (A1)
1 LEVEL (A2)
1 GNPGACTU (A8)
END-DEFINE
INPUT (SG=OFF) ERRNUMBER LINENUMBER STATUS GNPGACTU LEVEL
IF ERRNUMBER=6296
  STACK TOP COMMAND "FIN"
END-IF
END
```

Restrictions

There are several restrictions when using the Entire Screen Builder viewers with Natural applications on UNIX and OpenVMS hosts:

- **Runtime errors in Natural applications**
  Runtime errors in Natural UNIX and Natural OpenVMS applications are not handled correctly by Entire Screen Builder. This may lead to a loss of the session. Bypass: use the Natural system variable `*ERROR-TA` to handle the error. See the Natural documentation for details. Sample Natural error transaction:

  ```
  DEFINE DATA
  LOCAL
  1 ERR_INFO
    2 ERR_NR(N5)
    2 ERR_LINE(N4)
    2 ERR_STAT(A1)
    2 ERR_PNAM(A8)
    2 ERR_LEVEL(N2)
  END-DEFINE
  INPUT ERR_INFO
  DISPLAY ERR_INFO
  STACK TOP COMMAND 'E'
  END
  ```

  Starting with Natural 6.1, a default error transaction named `NSWUCIET` will be used if Natural is running with Entire Screen Builder and an error transaction has not been set by the user.

- **Return to the Natural main screen**
  You cannot use Natural applications that return to the Natural main screen. This always leads to wrong screen display and a loss of the session.

- **Natural editors and utilities**
  You cannot use the Natural utilities such as SYSMAIN, SYSDDM and editors such as NATEDIT. This always leads to wrong screen display and a loss of the session.

- **Natural system commands**
  You cannot use any Natural system command such as `CATALL`, `FIND`, `GLOBALS`, `HELP`, `KEY`, `LIST`, `SCAN` or `XREF`. This always leads to wrong screen display and a loss of the session.

- **Natural commands SETUP and RETURN**
  You should not use the Natural commands `SETUP` and `RETURN` as this may lead to a loss of the session.

- **Terminal commands**
  Terminal (%) commands are not supported. They do not work when entered in a viewer.

- **Internal REINPUT**
  The error messages of an internal `REINPUT` are not displayed in the viewers (for example, if you enter the number 500 into a I1 field: NAT1142). Examples of such messages:

NAT1142	Input results in integer value overflow.
NAT1125	Too many significant digits in numeric input value.
NAT1143	Input does not correspond to input edit mask.
NAT1011	Requested function key not allocated.

- **Natural system variable `*INIT-ID`**
  When using the Entire Screen Builder viewers with Natural applications on OpenVMS hosts, the Natural system variable `*INIT-ID` will not be filled with a value for the terminal type. Instead, it will contain the value `notty`.

HTTP Tunneling

The HTTP tunneling mechanism is frequently used in the Internet environment. It is used for sending and receiving the packets through the Web server (to traverse firewalls and/or proxies).

Entire Screen Builder supports tunneling with the following HTTP servers: Microsoft Internet Information Server and Apache Web Server.

This chapter covers the following topics:

- Setting Up Tunneling
- Customizing the Viewers

Notes:

1. The Microsoft Internet Explorer will limit the number of simultaneous connections that it will make to a single HTTP server. If you exceed this limit, the requests will block until one of the current connections has completed. For more details, see *Microsoft Knowledge Base Article - Q183110* at *http://support.microsoft.com/support/kb/articles/q183/1/10.asp*.
2. If you want to enable tunneling for the standalone version of the Terminal Viewer, see *Defining the Entire Screen Builder Server* in the *Terminal Viewer* documentation.
3. The XML Version can only be used with the ISAPI DLL. See *Configuring the XML Version* for detailed information.

Setting Up Tunneling

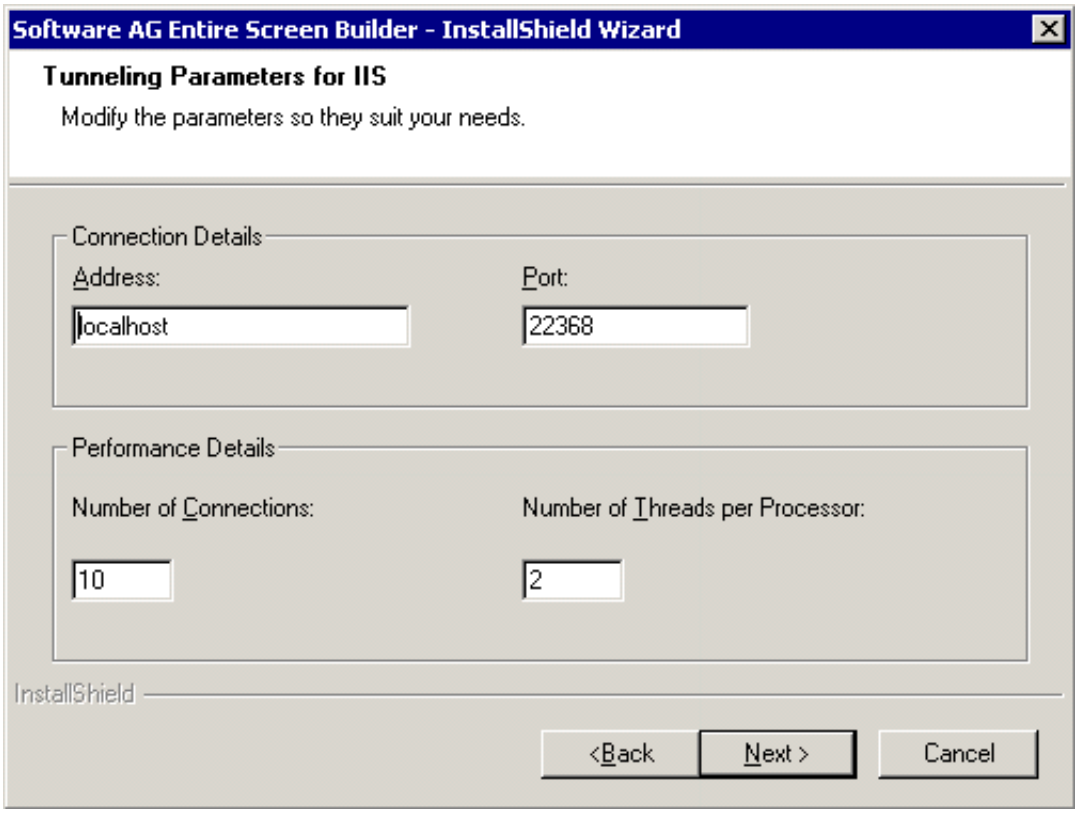
For tunneling, the Entire Screen Builder server extensions have to be installed for the HTTP server that you want to use. It is possible to use an HTTP server that is located on another machine (i.e. not on the machine on which the Entire Screen Builder Server has been installed).

The following topics describe how to set up tunneling for the different HTTP servers:

- Microsoft Internet Information Server under Windows
- Apache Web Server 2.0 under Windows
- Apache Web Server 1.3 and 2.0 on Solaris

Microsoft Internet Information Server under Windows

When you install Entire Screen Builder it is also possible to install the extensions for the Internet Information Server. The following dialog appears during installation when you have chosen to install this feature (custom installation):



Note:
If you want to change any values provided in this dialog box after installation, you have to change the corresponding parameters in the *Tunneling.reg* file. This file can be found in the folder *\Program Files\Software AG\Entire Screen Builder 5\server extensions\tunneling*. To import the new values into your registry, you have to execute (double-click) the modified *Tunneling.reg* file.

Specify the following information:

Address

Specify the IP address or host name of the machine on which the Entire Screen Builder Server is running.

Port

Specify the port number.

Important:
For address and port, specify the same information that has been defined in the System Management Hub. See *Tunneling Server* in Entire Screen Builder's *System Management Hub* documentation.

Number of Connections

Specify the number of connections in the pool.

Number of Threads per Processor

Specify the number of worker threads to handle the communications to hosts and clients and to generate screens. This value should be increased when the responsiveness of the server drops (i.e. the time for a new screen to be processed for a client increases but the server processor time as seen in the Windows Task Manager remains low).

The following options are intended for problem analysis. They can only be changed directly in the *Tunneling.reg* file and should only be used under supervision of your technical support.

TraceDir

Specify the trace directory.

EnableTrace

Specify 1 to enable traces, or 0 to disable the traces. The trace directory must exist before this parameter can be set to 1.

Web Viewer

The following steps are required if you plan to use Entire Screen Builder's Web Viewer:

1. In your server's home directory (usually, this is *C:\Inetpub\wwwroot*), create the logical name (alias) "ESBWebviewer".
2. Copy all files from the program folder ..*Entire Screen Builder 5\web viewer* to the folder linked to the logical name that was created in the previous step.

Windows Viewer

For Entire Screen Builder's Windows Viewer, no setup steps are required.

Terminal Viewer (Browser Version)

The following steps are required if you plan to use Entire Screen Builder's Terminal Viewer:

1. In your server's home directory (usually, this is *C:\Inetpub\wwwroot*), create the logical name (alias) "ESBTerminalviewer".
2. Copy all files from the program folder ..*Entire Screen Builder 5\terminal viewer* to the folder linked to the logical name that was created in the previous step.

Apache Web Server 2.0 under Windows

Copy the following files from the *Windows\Server Extensions\Tunneling\Apache* folder of the Entire Screen Builder CD to your Apache server *modules* folder:

- *APIERR.dll*
- *Esbap2api.dll*
- *mtcutil.dll*
- *mtcmemorypool.dll*
- *NswMessages.dll*
- *mtctrace.dll*

Modify the Apache Web Server configuration file *httpd.conf* as follows:

1. Set the directive KeepAlive to On.
2. Set the directive KeepAliveTimeout to a value greater than 20.

   This directive corresponds to Entire Screen Builder's tunneling poll time. See *Overview of Client Control Properties* in the *User Exits* documentation.
3. Set the directive MaxKeepAliveRequests to 0 (zero) to allow an unlimited amount of requests during one session.
4. Add the following lines at the end:

```
LoadModule esb_module modules/esbap2api.dll
<Location /esbhandler>
   EsbParms IP_ESB_server port_number_ESB_server
   TracesParms traces_directory 0/1
   SetHandler esb-handler
</Location>
```

   The first directive loads the DLL (or shared library on UNIX).

   The second directive defines the following parameters:

   `EsbParms`

   - Specify the IP address or host name of the machine on which the Entire Screen Builder Server is running.
   - Specify the port number.

   For the above IP and port values, specify the same information that has been defined in the System Management Hub. See *Tunneling Server* in Entire Screen Builder's *System Management Hub* documentation.

`TracesParms`

- These parameters are intended for problem analysis. They should only be used under supervision of your technical support.

  Specify the trace directory, and set the flag either to 1 to enable traces, or to 0 to disable the traces. The trace directory must exist before the flag can be set to 1.

There must not be a folder named "esbhandler" below your *document-root*.

5. Optionally - translate the name of your image folder by adding an Alias directive like the following:

   ```
   Alias /name_of_image_reposistory/ "full_path_to_your_images/"
   ```

   You have to add the above Alias directive after the following:

   ```
   # Aliases: Add here as many aliases as you need (with no limit). The format is
   # Alias fakename realname
   ```

Web Viewer

The following steps are required if you plan to use Entire Screen Builder's Web Viewer:

1. Create a folder called *ESBWebviewer* below your *document-root*.
2. Copy all files from the folder ..*Entire Screen Builder 5\web viewer* to the new folder.

Windows Viewer

For Entire Screen Builder's Windows Viewer, no setup steps are required.

Terminal Viewer (Browser Version)

The following steps are required if you plan to use Entire Screen Builder's Terminal Viewer:

1. Create a folder called *ESBTerminalviewer* below your *document-root*.
2. Copy all files from the folder ..*Entire Screen Builder 5\terminal viewer* to the new folder.

Apache Web Server 1.3 and 2.0 on Solaris

When the Entire Screen Builder Server is installed on a Solaris machine, take the files from the *unix\sun\serverextensions\apache* folder of the Entire Screen Builder CD. This folder contains further folders: *32bits* and *64bits*.

When using the 32 bits version of Apache, copy the following files from the *32bits* folder to your Apache Web Server *libexec* directory:

- *libapierr32.so*
- *libesbap2api32.so* (for version 2.0 only)
- *libesbapapi32.so* (for version 1.3 only)
- *libesbnsapi32.so*
- *libmtcmemorypool32.so*
- *libmtcmessages32.so*
- *libmtctrace32.so*
- *libmtcutil32.so*

When using the 64 bits version of Apache, copy the following files from the *64bits* folder to your Apache Web Server *libexec* directory:

- *libapierr.so*
- *libesbap2api.so* (for version 2.0 only)
- *libesbapapi.so* (for version 1.3 only)
- *libesbnsapi.so*
- *libmtcmemorypool.so*
- *libmtcmessages.so*
- *libmtctrace.so*
- *libmtcutil.so*

Add your Apache Web Server *libexec* directory to your LD_LIBRARY_PATH environment variable.

Modify the configuration file *http.conf* as described above for the Windows platform, but use the extension for shared libraries for Solaris:

```
LoadModule esb_module modules/filename
```

where *filename* is to be replaced with one of the following:

- *libesbap2api32.so* for version 2.0, 32, bits
- *libesbapapi32.so* for version 1.3, 32 bits
- *libesbap2api.so* for version 2.0, 64, bits
- *libesbapapi.so* for version 1.3, 64 bits

Web Viewer and Terminal Viewer (Browser Version)

Copy the Entire Screen Builder viewers as described above for the Windows platform.

Customizing the Viewers

When tunneling is used, server address and port number are not used by the GUI viewers. Any defined values are ignored.

The values for the tunneling type are:

M if a Microsoft Internet Information Server is used (with ISAPI).

A if an Apache Web Server is used.

The tunnelling poll time is the time in seconds the viewer polls the tunneling server for new data (asynchronous messages and screens).

Normally, the HTTP port is 80, or 440 for HTTPS.

How to enable tunneling for the different viewers is described in the topics below:

- Windows Viewer
- Web Viewer
- Terminal Viewer (Browser Version)

A detailed description of the properties mentioned below can be found in the *Overview of Client Control Properties* which is part of the *User Exits* documentation.

Windows Viewer

The following steps are required to configure Entire Screen Builder's Windows Viewer:

1. On your web server, create a logical name (alias) for the location in which the images are to be stored (usually "ESB_Repository").
2. Invoke the Windows Viewer. See *Invoking the Windows Viewer* in the *GUI Viewers* documentation.
3. In the resulting dialog box, select a connection and choose the **Edit** button.

   The Client Control Properties dialog box appears.
4. Select the HTTP page.
5. Specify the HTTP server address and the HTTP server port used for tunneling.
6. Make sure that the **HTTP tunneling** check box is selected.
7. Choose the **OK** button.

Web Viewer

Add the following entries to your customized HTML pages for the end user (for example, *ExampleEndUserPage.htm* which is provided in the Entire Screen Builder folder *web viewer*):

```
<param name="HTTPSERVER" value="Name_or_IpAddress_of_HttpServer">
<param name="HTTPPORT" value="PortNumber">
<param name="TUNNELING" value="1">
<param name="TUNNELINGTYPE" value="M">
<param name="TUNNELINGPOLLTIME" value="20">
```

See also: *Customizing the Web Viewer*.

Terminal Viewer (Browser Version)

Add the following entries to your customized HTML pages for the end user (for example, *ExampleTerminalViewer.htm* which is provided in the Entire Screen Builder folder *terminal viewer*):

```
<param name="SERVERNAME" value="Name_or_IpAddress_of_HttpServer">
<param name="PORTNUMBER" value="PortNumber">
<param name="USEHTTPCONNECTION" value="1">
<param name="TUNNELINGTYPE" value="M">
<param name="POLLTIMEOUT" value="20">
```

See also: *Customizing the Terminal Viewer*.

Configuring the XML Version

This chapter covers the following topics:

- About the XML Version
- Browser Restrictions
- Basic Requirements for the XML Version
- Required Folders on the Internet Information Server
- Configuring the Internet Information Server
- Configuring the ISAPI DLL
- Customizing the XSL Files

About the XML Version

Prerequisites

The XML Version of Entire Screen Builder requires Microsoft Internet Information Server (IIS) running on Windows 2000 or above. In addition, version 3.0 of the MSXML DLL must be installed on the server machine running the Internet Information Server. This can be downloaded from the Microsoft web site, and is most easily installed by upgrading this computer to run Internet Explorer 6.0.

The client browsers must be capable of running JavaScript and displaying frames.

Components

The XML Version of Entire Screen Builder consists of two separate components.

The first component is the server module which is loaded by the Entire Screen Builder Server and is initialized on installation of the product. This can be configured using the System Management Hub. See *XML Version* in Entire Screen Builder's *System Management Hub* documentation.

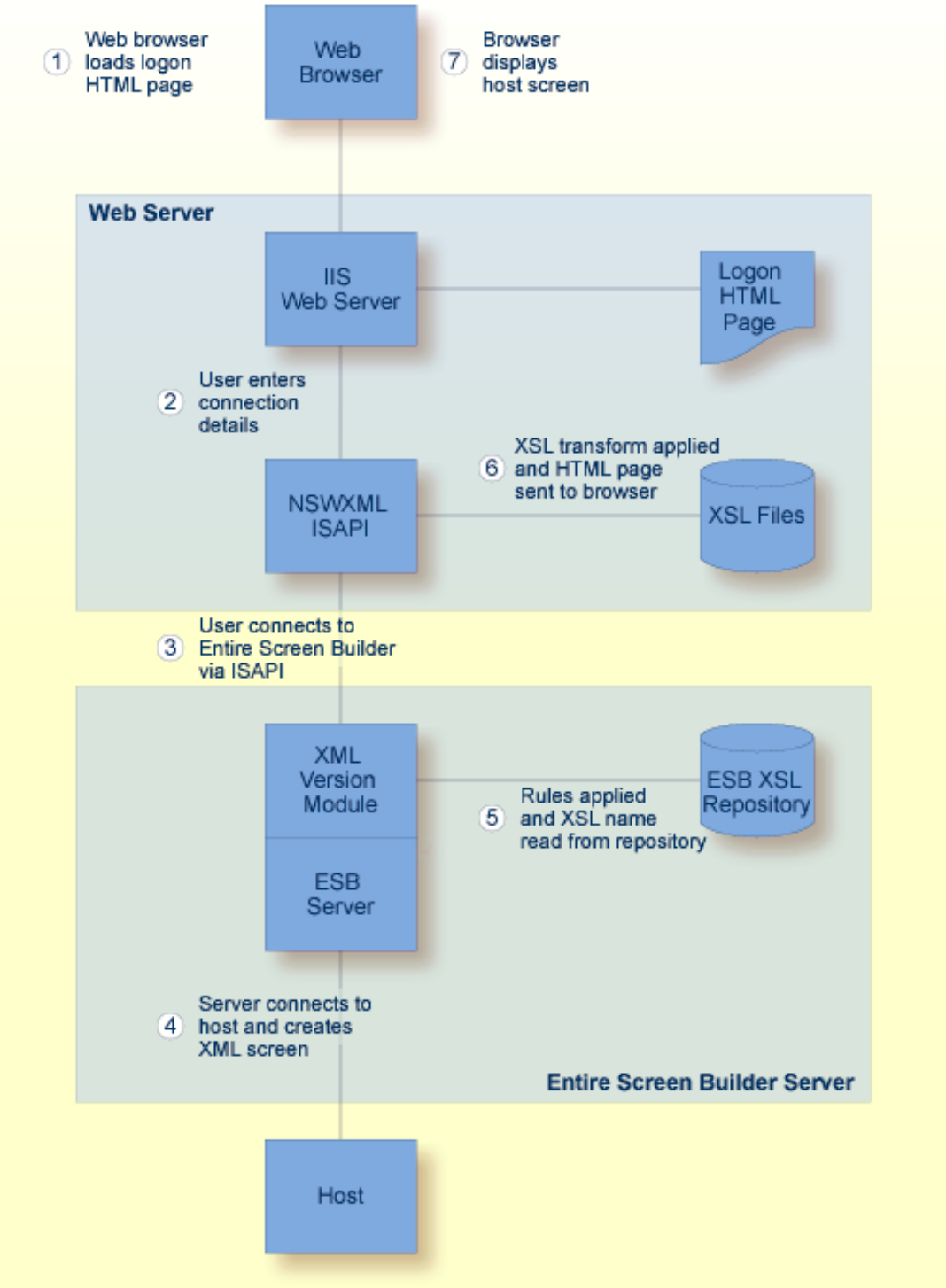
The second component is an Internet Information Server ISAPI DLL. This must be configured to run on Windows 2000 or above. It is planned in future versions to provide server extensions for Apache web servers.

Making a Connection

In order to use the XML Version from Entire Screen Builder, each user must connect to the server via an extension DLL (ISAPI) running inside Microsoft Internet Information Server.

This ISAPI DLL handles the connection to the Entire Screen Builder server, and converts the XML output to HTML.

The following graphic illustrates the control flow for a connection:



Browser Restrictions

Netscape/Mozilla

We have detected problems running all versions of Netscape Navigator when using the PF keys from the keyboard. On versions 4.x only the Carriage Return key can be identified, all other PF keys are not passed to the keyboard handler and the toolbars must be used.

Also note that the automatic reload on size function of Netscape 4.x is not supported since the reload function causes the host connection to be broken. Therefore, we recommend that you either block access to Netscape 4.x or only allow users to connect with a non-resizable browser window.

On version 7.0 of Netscape, the function keys such as PF5 are not intercepted completely and are returned to the browser. This fires a reload event which may cause the session to the host to be lost.

Others

On other browsers such as Opera the JavaScript implementation does not allow cross-frame scripting. Therefore, the no-frames option must be used (for further details, see the document *ConfigureCustomizeXMLVersion.doc* that is included as part of the preview kit provided in the *windows\Server Extensions\SDK_Preview* folder on the Entire Screen Builder CD).

Basic Requirements for the XML Version

Not all steps are required for the most basic installation. Optional steps are marked.

Step One - Install the Entire Screen Builder Server

- Install the Entire Screen Builder Server: Production Version, XML Version (EMV).
- Configure the Entire Screen Builder Server using the System Management Hub:
    - Create one or more host sessions. See *Host Sessions* in Entire Screen Builder's *System Management Hub* documentation.
    - Optional: set the listen port for the XML Version (if the default is not suitable). See *XML Version* in Entire Screen Builder's *System Management Hub* documentation.

Step Two - Connect

- Browse to the website, for example *www.mydomain.com/esb/esblogon.htm*.
- Enter the connection ID and any user details you require.
    - If no users are defined in Entire Screen Builder, you should use the Anonymous logon.
    - Enter the connection ID of the host session you have created.
- Choose **Connect**. The first host screen will appear.

Required Folders on the Internet Information Server

Folders for the following types of files are required on the Internet Information Server:

XSL Files

This folder contains all XSL files you have defined for the application (at least, the *default.xsl* provided). This folder must be accessible from the ISAPI DLL. It must have the same structure as the repository to which it refers.

HTML Files

This folder contains all HTML files for the ISAPI DLL. This folder must be accessible from the ISAPI DLL. It must have the same structure as the repository to which it refers.

This folder must contain the following files:

- *esbframebase.htm*

  Contains the base layout of the HTML page from Entire Screen Builder.

- *esbdisconnect.htm*

  Contains the page shown when the host has disconnected.

- *esberror.htm*

Configuring the Internet Information Server

The installation process for the XML ISAPI extensions adds the following virtual directories:

- *esbscripts*

  The ISAPI extension DLLs are placed here. This virtual directory is given "Script and Executable" access.

- *esbxsl*

  This gives access to XSL files when the XSL transformation is defined as being performed by the browser and not by the ISAPI extension (for further details, see the document *ConfigureCustomizeXMLVersion.doc* that is included as part of the preview kit provided in the *windows\Server Extensions\SDK_Preview* folder on the Entire Screen Builder CD).

- *esb*

  This virtual directory references the folder *\Program Files\Software AG\Entire Screen Builder 5\server extensions\xml\html* and contains an example logon page, *esblogon.htm*.

  For example, after installation you can browse to *http://myserver/esb/esblogon.htm* to test the XML Version.

Configuring the ISAPI DLL

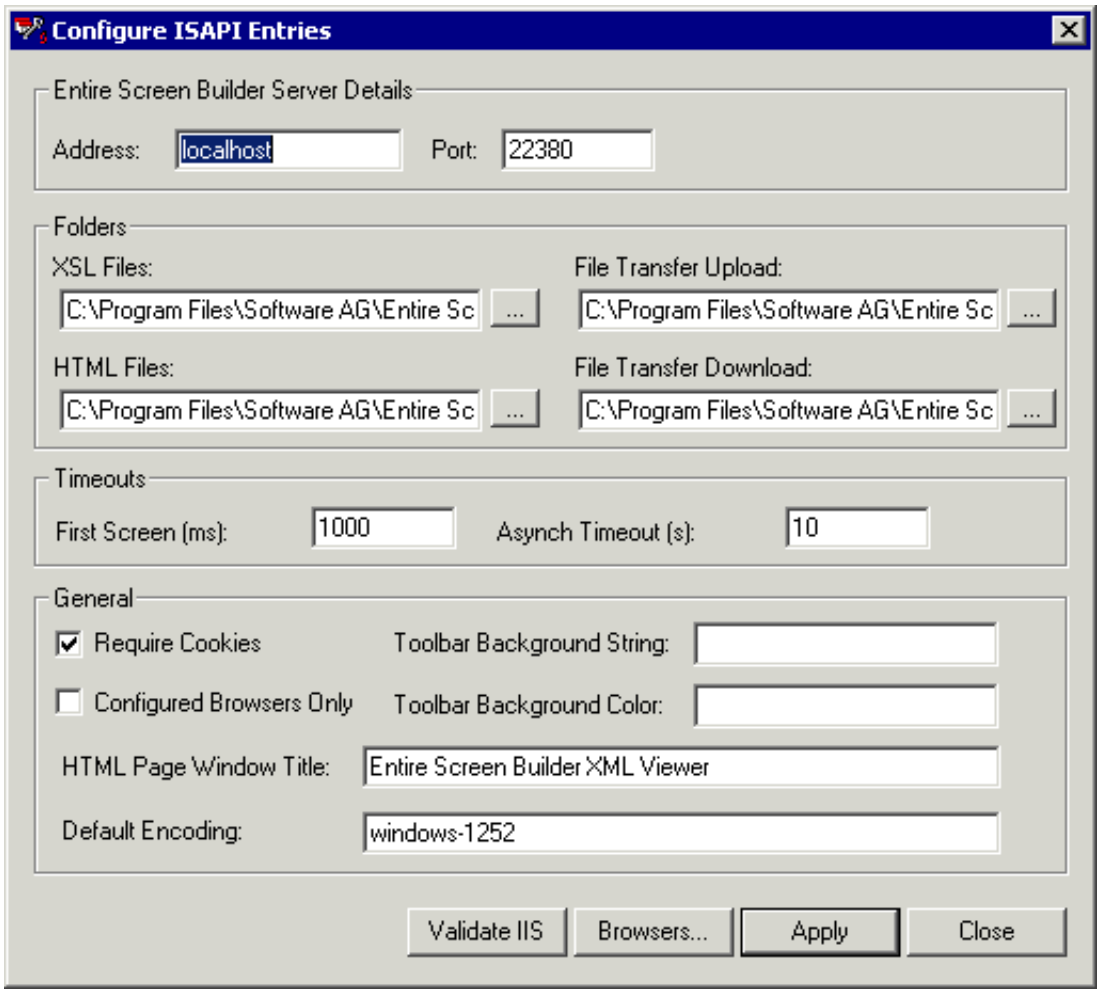
The XML Configurator for the ISAPI DLL can be used to modify the registry settings for the Internet Information Server. It must be run on the same computer as the Internet Information Server is located.

Important:
It is strongly recommended that you use SSL in the connection between the browser and web server when on the Internet. Otherwise, all data transferred is insecure and there is a risk of this data being accessed by a third party. This risk includes all host passwords.

To configure the ISAPI DLL

1. From the Start menu, choose **Programs > Software AG Entire Screen Builder *n.n.n* > Server Extensions > XML Configurator**.

   The Configure ISAPI Entries dialog box appears.



   Note:
   The above dialog box may also appear during installation. When you install the extensions for the Internet Information Server for the XML Version, a dialog appears asking whether you want to configure the ISAPI DLL.

2. Modify all required options as described in the *Overview of Options* below.

   For a description of the **Validate IIS** button, see *Defined Websites* below.

   For a description of the **Browsers** button, see *Browser Configuration* below.

3. Choose the **Close** button to save the new settings.

4. Restart the Internet Information Server to load the new settings.

Overview of Options

Address

The IP address or host name of the machine on which the Entire Screen Builder Server is running.

Port

The administration port number (listen address) that has been defined in the System Management Hub. See *XML Version* in Entire Screen Builder's *System Management Hub* documentation.

XSL Files

The full path to the XSL repository on the Internet Information Server machine. For further details, see the documentation included as part of the preview kit provided in the *windows\Server Extensions\SDK_Preview* folder on the CD.

HTML Files

The path to the folder containing the required HTML template files.

Required files:

- *esbframe.htm*
- *esbdisconnect.htm*
- *esberror.htm*

  This page is shown when an error has occurred and the host session cannot continue. You should include the string "!!!ESB ERROR INSERTION!!!" in this file as a placeholder for any Entire Screen Builder error message.

File Transfer Upload

The path to the temporary folder to be used for data uploads. If this folder does not exist, or if this text box does not contain an entry, data uploads are disabled.

File Transfer Download

The path to the temporary folder to be used for data downloads. If this folder does not exist, or if this text box does not contain an entry, data downloads are disabled.

First Screen

The length of time between the browser connecting and requesting the first screen. If an application takes longer to start, this timeout can be increased. Timeout is in milliseconds.

Asynch Timeout

The length of time that the browser waits before checking for new screens or disconnections returned by the host. If an application has many asynchronous screens, this timeout can be lowered, but this will increase the amount of server traffic. Timeout is in seconds.

Require Cookies

The Entire Screen Builder Server attempts to set a unique cookie to enhance security and identify connected sessions uniquely. However, if the browser connecting does not allow cookies to be set, the connection will still be enabled unless this check box is enabled. This option is recommended, and if used in conjunction with an SSL connection provides a high degree of data protection.

Configured Browsers Only

Normally, all browsers that attempt to connect to the server will be allowed, unless specifically blocked in the browser configuration dialog. If you use specific features that are only available on a limited number of browsers, you can use this option to stop other users from connecting.

Toolbar Background String / Toolbar Background Color

The XML Version uses several generated HTML screens for actions such as data transfer. It also uses the toolbar which appears at the top of the screen. If you are using a certain background color or image file, you can specify one or the other here. This will then be added to the HTML file, for example:

```
<BODY background="/images/background.gif">
```

or

```
<BODY bgcolor="#ffffff">
```

If both fields are left blank, only `<BODY>` is used.

If you have defined a background color in the host session, you can use the `bgcolor` entry to keep the complete screen the same color.

HTML Page Window Title

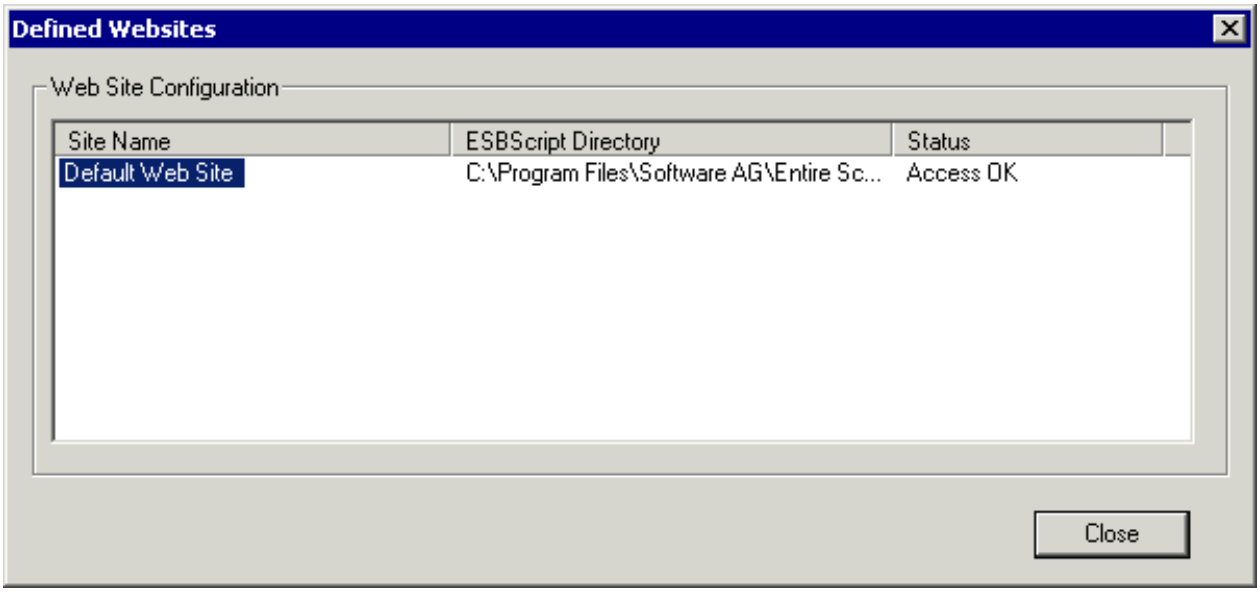
This text will be used for the <TITLE> section of any generated HTML pages.

Default Encoding

You can specify another default encoding.

Defined Websites

When you choose the **Validate IIS** button in the Configure ISAPI Entries dialog box, the Defined Websites dialog box appears.

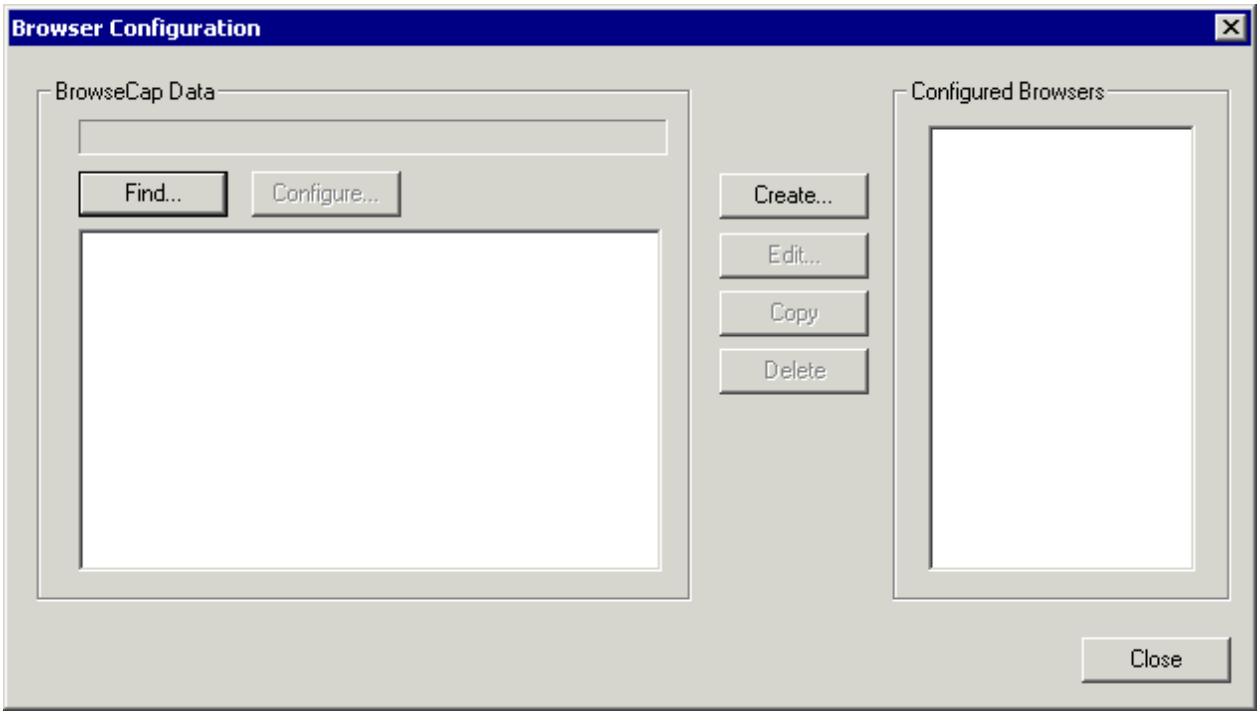


This dialog box shows the current location of the *esbscripts* directory. It also shows whether the permissions are sufficient for the XML Version. If the Status column shows "Too few permissions", you must set the Execute Permissions for this virtual directory to "Scripts and Executables" in the Internet Information Server Manager.

Note:
These permissions are set by default on installation.

Browser Configuration

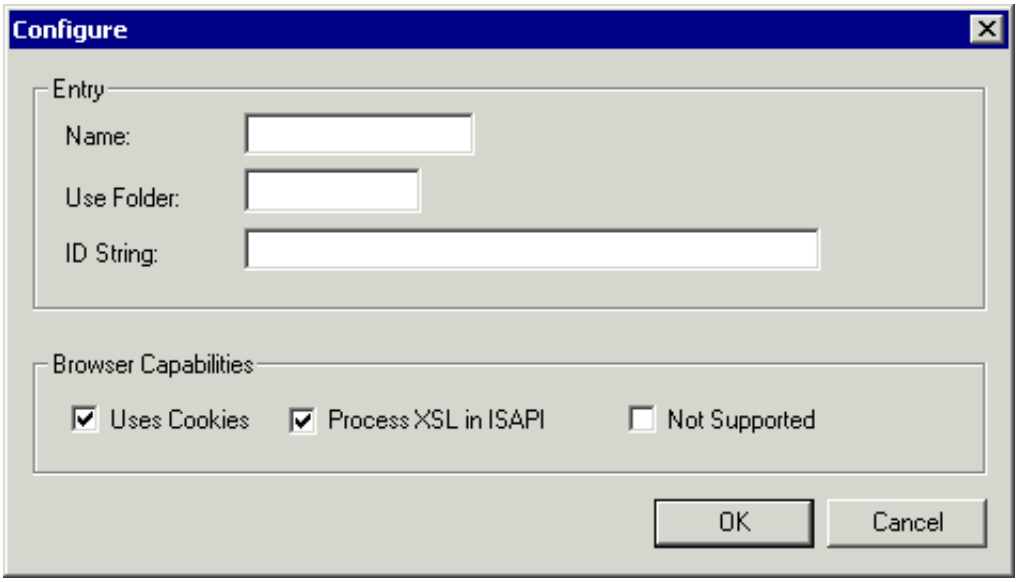
When you choose the **Browsers** button in the Configure ISAPI Entries dialog box, the Browser Configuration dialog box appears.



You can now specify XSL files and parameters. You can also block the use of certain browsers.

You can use a *browsecap.ini* file to get the default entries for the ID string that is used in the Configure dialog box below. This file is provided in the *inetsrv* folder of your *Windows\System32* folder. Choose the **Find** button to define this file.

To configure a browser, choose the **Create** button in the above Browser Configuration dialog box. The Configure dialog box appears.



Specify the following information:

Name

Name of the browser.

Use Folder

This is the relative path to the name of the folder containing the XSLT files that you have customized for this browser. If this field is blank, the default folder will be used.

A specific XSLT file for Netscape is included with Entire Screen Builder and in that case this value can be set to "Netscape" for Netscape browsers.

ID String

This field contains the browser agent description. In the cases where the same browser has various user-agent strings depending on the operating system, you can set this field to the start of the user-agent string. For example, the following string will identify all Internet Explorer 6 versions:

```
Mozilla/4.0 (compatible; MSIE 6
```

If you wish to identify browsers where the user-agent differs at the end (for example Netscape and Mozilla), you can use the asterisk (*) wildcard character. For example, the following string will uniquely identify Mozilla on Windows operating systems:

```
Mozilla/5.0 (Windows;*Gecko/20020910
```

Uses Cookies

If this check box is cleared, Entire Screen Builder will not attempt to set a connection cookie for this browser. Note that if the **Require Cookies** setting is active, this will block users of this browser.

Process XSL in ISAPI

If this check box is cleared, Entire Screen Builder will send XML to the browser. The browser must be capable of performing the XSL transformation itself. In this case, you must have defined the virtual directory *esbxsl*.

It is recommended that this check box is left checked unless you need access to the XML file in order to develop new XSLTs.

Please note that the ISAPI extension for Entire Screen Builder uses the MSXML3 library from Microsoft to execute these style sheet transformations. Therefore, it is recommended that you use Internet Explorer 6.0 or higher in this case.

Not Supported

This check box can be set to block all users of the specified browser from accessing Entire Screen Builder.

When you choose the **OK** button, the name of the browser appears in the Configured Browsers list box. The command buttons **Edit**, **Copy** and **Delete** can be used to manage the list of configured browsers.

When you have chosen the **Find** button, you can also select a browser in the list box below the **Find** button and then choose the **Configure** button. This displays the Configure dialog box (see the above description) with all information for this browser. When you choose the **OK** button in the Configure dialog box, the browser is added to the Configured Browsers list box.

Customizing the XSL Files

For further details, see the document *ConfigureCustomizeXMLVersion.doc* that is included as part of the preview kit provided in the *windows\Server Extensions\SDK_Preview* folder on the Entire Screen Builder CD.